ARMY RESEARCH LABORATORY

Projectile Base Bleed Technology
Part II: User's Guide
CMINT Computer Code
Version 5.04-BRL

Howard J. Gibeling
Richard C. Buggeln

ARL-CR-3 November 1992

prepared by

Scientific Research Associates, Inc.
50 Nye Road
P.O. Box 1058
Glastonbury, CT 06033

under contract

DAA15-88-C-0040

NOTICES

Destroy this report when it is no longer needed. DO NOT return it to the originator.

Additional copies of this report may be obtained from the National Technical Information Service, U.S. Department of Commerce, 5285 Port Royal Road, Springfield, VA 22161.

The findings of this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

The use of trade names or manufacturers' names in this report does not constitute indorsement of any commercial product.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	November 1992	Final, July 1988 - July 1991

4. TITLE AND SUBTITLE

PROJECTILE BASE BLEED TECHNOLOGY; PART II: USER'S GUIDE CMINT COMPUTER CODE, VERSION 5.04-BRL

5. FUNDING NUMBERS

DAA15-88-C-0040

6. AUTHOR(S)

HOWARD J. GIBELING and RICHARD C. BUGGELN

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Scientific Research Associates, Inc.
50 Nye Road, P.O. Box 1058
Glastonbury, CT 06033

8. PERFORMING ORGANIZATION REPORT NUMBER

R91-930020-F

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

U.S. Army Research Laboratory
ATTN: AMSRL-OP-CI-B (Tech Lib)
Aberdeen Proving Ground, MD 21005-5066

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

ARL-CR-3

11. SUPPLEMENTARY NOTES

The Contracting Officer's Representative for this report is Charles J. Nietubicz, U.S. Army Research Laboratory, ATTN: AMSRL-WT-PB, Aberdeen Proving Ground, MD, 21005-5066.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Detailed finite rate chemistry models for H_2 and H_2-CO combustion have been incorporated into a Navier-Stokes computer code and applied to flow field simulation in the base region of an M864 base burning projectile. Results without base injection were obtained using a low Reynolds number k-ε turbulence model and several mixing length turbulence models. The results with base injection utilized only the Baldwin-Lomax model for the projectile forebody and the Chow wake mixing model downstream of the projectile base.

A validation calculation was performed for a supersonic hydrogen-air burner using an H_2 reaction set which is a subset of the H_2-CO reaction set developed for the base combustion modeling. The comparison with the available experimental data was good, and provides a level of validation for the technique and code developed. Projectile base injection calculations were performed for a flat base M864 projectile at $M_\infty = 2$. Hot air injection, H_2 injection and H_2-CO injection were modeled, and computed results show reasonable trends in the base pressure increase (base drag reduction), base corner expansion and downstream wake closure location.

14. SUBJECT TERMS

Projectile Base Combustion
Navier-Stokes Base Flow Analysis
Hydrogen Combustion
Combustion
Hydrogen-Carbon Monoxide Combustion
Projectiles
Projectile Base Drag

15. NUMBER OF PAGES

95

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

INTENTIONALLY LEFT BLANK.

TABLE OF CONTENTS

Intentionally left blank.

LIST OF FIGURES

LIST OF TABLES

Intentionally left blank.

INTENTIONALLY LEFT BLANK.

PREFACE

The U.S. Army Ballistic Research Laboratory was deactivated on 30 September 1992 and subsequently became a part of the U.S. Army Research Laboratory (ARL) on 1 October 1992.

1.0 INTRODUCTION

This User's Manual describes the SRA CMINT computer code, which solves the compressible time-dependent Navier-Stokes equations in both two and three space dimensions using a linearized block implicit solution algorithm with Douglas-Gunn splitting. The code is written for general nonorthogonal coordinate transformations with a moving coordinate option and employs centered spatial difference formulas with adjustable dissipation. The code employs matrix conditioning techniques to accelerate convergence to steady solutions. The code has an optional low Mach number formulation of the equations for computing near-incompressible flows. Both k-ϵ and algebraic mixing length turbulence models are available, with resolution of viscous sublayer regions. The code allows a wide variety of boundary conditions, including cascade periodicity, and all boundary conditions are treated implicitly within the numerical solution procedure. The code has a generalized multiple-corner capability and is fully vectorized.

This User's Manual is meant to serve as a guide in helping the user make successful runs with the SRA CMINT series of computer programs. The CMINT code will solve the governing Navier-Stokes equations, subject to the boundary conditions for a large variety of geometric configurations. The code is constructed so that many necessary input items are set by default. These include boundary conditions, turbulence model constants, time step control parameters, etc. It is suggested that the novice user maintain default values unless problems arise; changes to counteract possible problems are discussed in a later section. The more experienced user may desire to change some of these default parameters and may even want to modify boundary conditions, turbulence model, etc. Since the code is written in modular form, such modifications often can be made although successful modification does require an understanding of code details.

The User's Manual is divided into ten parts consisting of: (1) an introduction, (2) a brief description of each subroutine and its use, (3) a description of the PARAMETER statements used to dimension array variables in the CMINT code, (4) a list of the major FORTRAN variables and a description of their meaning, (5) a description of the logical file units utilized by the CMINT computer code, (6) a brief discourse on the general strategy for the running of the CMINT computer code, (7) a description of the grid generation method of Oh, (8) the protocol for grid point and boundary identification (9) a detailed

description of the CMINT computer code Namelist input, and (10) a description of the multiple species gas input (also see Appendix A). For details of the equations solved, boundary conditions utilized and the numerical procedure, the user is referred to Refs. 1-5.

2.0 DESCRIPTION OF SUBROUTINES

Subroutine	Purpose
ADIC	Primary control subroutine for the LBI-ADI procedure - called from subroutine EXEC.
ADICP1	Secondary control subroutine for first ADI sweep - called from subroutine ADICX.
ADICP2	Secondary control subroutines for second and third ADI sweeps - called from subroutines ADICY and ADICZ.
ADICTRL	Subroutine which sets up control parameters for the ADI subroutines - called from subroutine CORNER.
ADICX	Primary control subroutine for the first ADI sweep - called from subroutine ADIC.
ADICY	Primary control subroutine for the second ADI sweep - called from subroutine ADIC.
ADICZ	Primary control subroutine for the third ADI sweep - called from subroutine ADIC.
AMATRX	Primary control subroutine for the calculation of time and source terms of the governing partial differential equations - called from subroutines ADICP1, ADICP2 and BC.
ARTVIS	Subroutine to calculate artificial dissipation terms of the governing partial differential equations - called from subroutines GENEQ1 and GENEQ2.
ATIME	Subroutine to calculate the time rate of change terms of the governing partial differential equations - called from subroutines AMATRX and BC.
BC	Subroutine which sets up all boundary conditions - called from subroutines GENBC and SETBV.
BCSET	Subroutine which sets default boundary conditions - called from subroutine READA.
BLT	Subroutine which calculates the boundary layer thickness. Normally requires user modification - called from subroutine EXTBV.
CMUINIT	Subroutine which iteratively solves for C_μ for the Jones-Launder turbulence model - called from subroutines SETDKE and BC.
CMUMOD	Subroutine which calculates C_μ from the Jones-Launder turbulence model - called from subroutine VISCOS.

Subroutine	Purpose
CONVCT1	Subroutine which calculates the convection terms of the governing partial differential equations; applicable to transformed cartesian equations - called from subroutine GENEQ1.
CONVCT2	Subroutine which calculates the convection terms of the governing partial differential equations; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
CORNER	Primary control parameter subroutine - called from subroutine READA.
CYPOL	Subroutine which calculates nonstandard terms of the governing partial differential equation; applicable to cylindrical-polar equations - called from subroutine GENEQ2.
DELU1	Subroutine which calculates the dilation terms of the momenta equations; applicable to transformed cartesian equations - called from subroutine GENEQ1.
DELU2	Subroutine which calculates the dilation terms of the momenta equations; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
DIFF1	Subroutine which calculates the diffusion terms of the governing partial differential equations; applicable to transformed cartesian equations - called from subroutine GENEQ1.
DIFF2	Subroutine which calculates the diffusion terms of the governing partial differential equations; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
DIL1	Subroutine which calculates the dilation contribution to the stress work portion of the energy equation; applicable to transformed cartesian equations - called from subroutine GENEQ1.
DIL2	Subroutine which calculates the dilation contribution to the stress work portion of the energy equation; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
DIVDIS1	Subroutine which calculates $\nabla \cdot \nabla$ and D:D for transformed cartesian systems - called from subroutine EXTBV.
DIVDIS2	Subroutine which calculates $\nabla \cdot \nabla$ and D:D for transformed cylindrical-polar systems - called from subroutine EXTBV.
DOP	Primary control subroutine for generation of spatial derivatives of the governing partial differential equations - called from subroutines ADICP1 and ADICP2.
DOUBLE	Control subroutine for reentrant corner logic - called from subroutines ADICX, BC, EXTBV and SETBV.

Subroutine	Purpose
EOSDP1	Subroutine which calculates the pressure gradient terms of the momenta equations; applicable to transformed cartesian equations - called from subroutine GENEQ1.
EOSDP2	Subroutine which calculates the pressure gradient terms of the momenta equations; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
EOSDPDT	Subroutine which calculates the time derivative of pressure terms of the energy equation - called from subroutine ATIME.
EOSPBC	General subroutine for use in the construction of pressure boundary conditions - called from subroutine BC.
EOSTBC	General subroutine for use in the construction of temperature boundary conditions - called from subroutine BC.
EOSUP	Subroutine for the calculation of pressure and temperature (given enthalpy, density and velocity components) or density and enthalpy (given pressure, temperature and velocity components) - called from subroutines EXTBV and FLWFLD.
EOSVDP1	Subroutine which calculates the V•∇P contribution to the static enthalpy version of the energy equation; applicable to transformed cartesian equations - called from subroutine SORCH1.
EOSVDP2	Subroutine which calculates the V•∇P contribution to the static enthalpy version of the energy equation; applicable to transformed cylindrical-polar equations - called from subroutine SORCH2.
EXEC	Secondary control subroutine for the CMINT computer code - called from the main program.
EXTBV	Primary control subroutine for performing the ADI endcapping and the updating of the dependent and derived variables - called from subroutine EXEC and READA.
EXTENT	Subroutine which generates control parameters for third direction derivatives - called from subroutine READA.
FLWFLD	Primary control subroutine for the generation of initial conditions - called from subroutine EXEC.
GENBC	Primary control subroutine for the generation of boundary conditions within an ADI sweep - called from subroutines ADICP1 and ADICP2.
GENEQ1	Secondary control subroutine which generates the spatial derivatives for the governing partial differential equations; applicable to transformed cartesian equations - called from subroutine BC and DOP.

Subroutine	Purpose
GENEQ2	Secondary control subroutine which generates the spatial derivatives for the governing partial differential equations; applicable to transformed cylindrical-polar equations - called from subroutine BC and DOP.
GSUPS	Subroutine which calculates the transformation for nonorthogonal boundary conditions - called from subroutine BC.
LEGVAR	General subroutine to check for an indefinite value of a variable - called from subroutines PRINT1 and WRPLOT.
LYNSYS	General matrix inversion subroutine - called from subroutine OHGRID.
MDPDT	Subroutine which calculates the convective-like term from the $\partial p/\partial t$ term of the energy equation with moving coordinates - called from subroutines GENEQ1 and GENEQ2.
MGAUSF	Generalized tridiagonal NxN block matrix inverter - called from subroutines ADICX, ADICY and ADICZ.
MGAUSP	Generalized tridiagonal NxN block matrix inverter with periodic boundary conditions - called from subroutines ADICP1 and ADICP2.
MGERR	Subroutine which calculates accuracy of matrix inversion - called from subroutines MGAUSF, MGAUSP, MGP1X1, MGS1X1, MGS2X2, MGS3X3, MGS4X4 and MGS5X5.
MGP1X1	Tridiagonal scalar matrix inverter with periodic boundary conditions - called from subroutines ADICX, ADICY and ADICZ.
MGS1X1	Tridiagonal scalar matrix inverter - called from subroutines ADICX, ADICY and ADICZ.
MGS2X2	Tridiagonal 2x2 matrix inverter - called from subroutines ADICX, ADICY and ADICZ.
MGS3X3	Tridiagonal 3x3 matrix inverter - called from subroutines ADICX, ADICY and ADICZ.
MGS4X4	Tridiagonal 4x4 matrix inverter - called from subroutines ADICX, ADICY and ADICZ.
MGS5X5	Tridiagonal 5x5 matrix inverter - called from subroutines ADICX, ADICY and ADICZ.
MINMAX	Subroutine which determines parameters which describe the type of finite difference molecule to be used for derivatives - called from subroutine BC.
MIXLEN	Subroutine which calculates the mixing length. Normally requires user modification - called from subroutine VISCOS.

Subroutine	Purpose
NORMD	Subroutine which calculates the distance to the nearest wall. Normally requires user modification - called from subroutine TIMGEO.
OHGRID	Subroutine which distributes grid points on a line by the method of Oh - called from subroutine TIMGEO.
PRESSET	Subroutine which saves the pressure at various locations for use in the formulation of boundary conditions. Normally requires user modification - called from subroutine EXTBV.
PRGEO	Control subroutine for the printing of geometric data - called from subroutine EXEC.
PRINT1	Subroutine which prints geometric and fluid dynamic data - called from subroutines PRGEO and PRNTF.
PRNTF	Control subroutine for the printing of fluid dynamic data - called from subroutine EXEC.
PRNTS	Subroutine which prints time step summary and convergence data - called from subroutine EXEC.
RDLIST, WRLIST	Subroutine which reads and writes NAMELIST data - called from subroutines READB and RESTRT.
READA	Primary control subroutine for the calculation of the control parameters, default parameters and variables and restart information - called from the main program.
READB	Control subroutine for the writing and reading of restart files - called from subroutine READA.
RESTRT	Subroutine which writes and reads restart files - called from subroutine READB.
SETBV	Control subroutine for performing the ADI endcapping - called from subroutine EXTBV.
SETDKE	Subroutine which generates initial conditions for the k-ϵ turbulence model - called from subroutine VISCOS.
SHIFT	Subroutine which shifts data; used for three-dimensional calculations with out-of-core option - called from subroutines ADIC, ADICZ, EXTBV, FLWFLD and TIMGEO.
SORCH1	Control subroutine for the calculation of nonstandard terms of the governing partial differential equations; applicable to transformed cartesian equations - called from subroutine GENEQ1.

Subroutine	Purpose
SORCH2	Control subroutine for the calculation of nonstandard terms of the governing partial differential equations; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
SOURCE	Subroutine which calculates source terms of the governing partial differential equations - called from subroutines AMATRX and BC.
SPREADTP	Subroutine which generates initial conditions for turbulent pipe flow - called from subroutine FLWFLD for ISPREAD = 1.
SPREAD2	Subroutine which generates initial conditions. Normally requires user modification - called from subroutine FLWFLD for ISPREAD = 2.
SSTST	Subroutine which calculates normalized changes of the dependent variables - called from subroutine EXTBV.
STRESSH1	Subroutine which calculates stress work terms of the energy equation; applicable to transformed cartesian equations - called from subroutine GENEQ1.
STRESSH2	Subroutine which calculates stress work terms of the energy equation; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
STRESSW1	Subroutine which calculates stress work terms of the energy equation; applicable to transformed cartesian equations - called from subroutine GENEQ1.
STRESSW2	Subroutine which calculates stress work terms of the energy equation; applicable to transformed cylindrical-polar equations - called from subroutine GENEQ2.
TANHYP	Subroutine which distributes grid points on a line by use of a hyperbolic tangent transformation - called from subroutine TIMGEO.
TIMGEO	Subroutine which calculates all geometric information - called from subroutine EXEC and READA.
TIMLFT	Subroutine which calculates amount of central processor time remaining - called from the main program and subroutine EXEC. (Not operational under the Unicos operating system.)
VISCOS	Subroutine which calculates the laminar and turbulent viscosities - called from subroutine EXTBV.
WALFLX	Subroutine which calculates the mass flux for wall blowing - called from subroutine BC.

Subroutine	Purpose
WHERE	Subroutine which determines the x and z section numbers and the nearest boundaries given the grid point - called from subroutine EXTENT, NORMDO and SETBV.
WRDISK	General purpose disk file writing and reading subroutine - called from subroutine WRSLAB.
WRPLOT	Subroutine which writes plot files - called from subroutine EXEC.
WRSLAB	Control subroutine for out-of-core disk reading and writing - called from subroutine ADIC, ADICZ, EXTBV, FLWFLD, PRINT1, READA, TIMGEO and WRPLOT.

Intentionally left blank.

3.0 PARAMETER STATEMENTS

The CMINT computer code makes extensive use of PARAMETER statements for dimensioning FORTRAN variables and for setting data statements. The primary advantage of PARAMETER statements is the ease with which dimensions of various arrays can be changed. The dimensions of many of the arrays in the CMINT code are interrelated and hence the redimensioning of the code is not a trivial matter. The use of PARAMETER statements allows these relationships to be coded in an analytic form. Thus the user needs to change only a few PARAMETER statements to redimension the code. Often it has been found that it is economical to have a version of the CMINT code dimensioned for a minimum amount of core required for a specific class of problems. This results in a minimum allocation of computer resources and hence computer charges and often results in improved job turnaround, as the priority algorithms are usually a function of requested storage. The CMINT code has primary PARAMETERS and derived PARAMETERS which will be described below. It should be emphasized at this point that if any of the PARAMETER statements are modified the entire program must be recompiled, as the use of PARAMETER statements results in a direct substitution of the value of a given PARAMETER at compilation time.

PARAMETER	PURPOSE
NALFMAX	Maximum number of metal mass fraction values for which equilibrium chemistry polynomial curve fits are stored. Default value is 21.
NBRKX	Maximum number of breaks in the boundary condition with respect to the x- direction. Default value is 10.
NBRKZ	Maximum number of breaks in the boundary condition with respect to the z- direction. Default value is 10.
NCLMX	Maximum number of cluster points for use with the method of Oh. Default value is 25.
NCOEQM	Maximum number of species (component) equations. Default value is 4.

PARAMETER	PURPOSE
NCPLD	Maximum number of coupled equations to be solved at one time. Default value is 5.
NCRMAX	Maximum number of reentrant corners allowed. Default value is 10.
NCRTMAX	Maximum number of computational corner points (both inward and REENTRANT). Default value is 40.
NDBLMX	Maximum number of airfoils for cascade rotor-stator domain. Used for JGRID = 4 only. Default value is 2.
NEXL	Maximum number of simultaneous ADI lines to be inverted. Default value is 61.
NEXLAM	Maximum number of simultaneous ADI lines to be inverted - used in subroutines MGS1X1, MGS2X2, etc. Must be a multiple of *NEXL*. Default value is NEXL.
NEXTX	Maximum number of extents on loops in the x- direction. Default value is 2.
NEXTZ	Maximum number of extents on loops in the z- direction. Default value is 2.
NGPTOT	Maximum number of grid points. For two-dimensional flow NGPTOT must be greater than or equal to NX*NZ. For three-dimensional flow NGPTOT must be greater than or equal to NX*NY*NZ for the code to be run in the in-core mode. For three-dimensional flow when NGPTOT is greater than or equal to NIN*MAX(NX*NY, NY*NZ) (where NIN is the number of planes in-core; default value of 5); (see $READ2) the code will be run in the out-of-core option. In this case, a storage unit must be assigned. Default value is 1.
NGVMAX	Maximum number of geometry variables. Default value is 26.

PARAMETER	PURPOSE
NLOOPMAX	Maximum number of loops for calculating z- direction derivatives. Default value is 10.
NPLVAR	Maximum numbers of plot variables for time-dependent plot history. Default value is 20.
NPOLY	Polynomial order for chemical equilibrium curve fits. Default value is 5.
NPTD	Size of the finite difference molecule. Default value is 3.
NREACM	Maximum number of Arrhenius chemical reactions. Default value is 11.
NSPECM	Maximum number of species. May be greater than NCOEQM. Default value is 11.
NSURMX	Maximum number of boundary surfaces. Default value is 10.
NTPLT	Maximum number of time steps for time history data storage. Default value is 1000.
NTRNGM	Maximum number of temperature ranges considered in the JANNAF thermochemical data curve fits. Default value is 3.
NVLAC	Maximum number of fluid and geometric variables. Default value is 36.
NX	Maximum dimension in x-direction. Default value is 100.
NXON	Maximum number of y^1 direction (LX) print control flags. Default value is 25.

PARAMETER	PURPOSE
NXONA	Maximum number of y^1 direction (LX) print control flags for averaging. Default value is 25.
NY	Maximum dimension in y-direction. Default value is 1.
NYON	Maximum number of y^2 direction (LY) print control flags. Default value is 25.
NYONA	Maximum number of y^2 direction (LY) print control flags for averaging. Default value is 25.
NZ	Maximum dimension in z-direction. Default value is 155.
NZON	Maximum number of y^3 direction (LZ) print control flags. Default number is 25.
NZONA	Maximum number of y^3 direction (LZ) print control flags for averaging. Default value is 25.

Additional parameters for the CELMINT Eulerian-Lagrangian two-phase flow version of the CMINT code are as follows:

NERFIN	Size of the inverse error function table for the random number generator. Default value is 50.
NINJMX	Maximum number of particle injection locations. Default value is 1.
NPARMX	Maximum number of computational particles. Default value is 1.

PARAMETER	DERIVED VALUE
NVPAR	Maximum number of variables in the particle property array (PAR). Default value is 13.
NVSAMP	Maximum number of particle sampling variables. Default value is 13.
NTRAMX	Maximum number of particle traces. Default value is 1.
NMOVMX	Maximum number of sub-time steps for each particle trace. Default value is 1.
NVTRAC	Maximum number of variables in the TRACE array. Default value is 1.
NWALMX	Maximum number of boundary surfaces in one direction. Default value is 1.
NEQS	Maximum external equation number to be solved. See Table 1. Default value is NCOEQM + 7.
NGRPMX	Maximum number of groups of coupled equations. Default value is NEQS.
NVSTOR	Maximum number of dependent variable coefficients to be stored for each equation. Default value is NEQS.
NXYMAX	MAX(NX,NY)
NXZMAX	MAX(NX,NZ)
NYZMAX	MAX(NY,NZ)
NYZMXP	MAX(NY,MIN((NY-1)*NZ+1,NZ))

PARAMETER	DERIVED VALUE
NN	MAX(NX,NY,NZ)
NSEGMAX	Maximum number of segments for ADI matrix inversion. Default value is (NN − 1)/NEXL + 1.
NTAUT	Maximum number of data points allowed for the taut spline curve fit routine TAUTSP. Default value is 2 * NN.
NYC	This must be set equal to NY for the cascade code options (JGRID 1, 2, 3 or 4). Default value is 1.
NZC	This must be set equal to NZ for the 3-D cascade code (JGRID = 1, 2, 3 or 4). Default value is 1.
NYZMXC	This must be set equal to NYZMAX for the cascade code options (JGRID = 1, 2, 3 or 4). Default value is 1.
NYZMXD	This should be set equal to NYZMXP for the cascade code options (JGRID = 1, 2, 3 or 4). Default value is 1.

4.0 LIST OF MAJOR FORTRAN VARIABLES

FORTRAN VARIABLE	DESCRIPTION
AD(NVLAC,NCRMAX,NY)	Storage for corner points
AINDEF	Indefinite value
ALPHA(NCLMX,3)	Bandwidth for method of Oh
AM(NCPLD,3*NCPLD+1,NEXLAM)	Storage for matrix inversion subroutines
AN(NVSTOR,NN)	Storage for time term coefficients
APR(NN,NN)	Storage for printing
AQ(NVLAC,NCRMAX,NY)	Storage for corner points
AVISC(3,NEQS)	Artificial dissipation parameters
BLPROF(NXZMAX,NYZMXP)	Boundary layer profile parameters
BM((3*NCPLD+1)*NCPLD,NEXLAM)	Storage for matrix inversion subroutines
C(NCPLD*NCPLD,NPTD,NEXL*NN)	Master storage for matrix elements
CH	Array index for inverse of specific heat (integer)
CLPX(NCLMX,3)	Grid point for method of Oh
CLPY(NCLMX,3)	Location for method of Oh
CLENG	Reference length
CM(NCPLD,3*NCPLD+1)	Storage for matrix inversion in subroutine SETBV
CMACH	Reference Mach number
CMUF(NN)	Prandtl-Kolmogorov coefficient
CNORM(NEXL*NN)	Storage for normalization in matrix inversion subroutines
CNORM1(NEXL*NN)	Storage for normalization in matrix inversion subroutines
CNORM2(NEXL*NN)	Storage for normalization in matrix inversion subroutines

FORTRAN VARIABLE	DESCRIPTION
CNORM3(NEXL*NN)	Storage for normalization in matrix inversion subroutines
CNORM4(NEXL*NN)	Storage for normalization in matrix inversion subroutines
CNORM5(NEXL*NN)	Storage for normalization in matrix inversion subroutines
COEF(3,3)	Coefficient storage for cylindrical-polar convection terms
COF1(NN)	Coefficient storage for cylindrical-polar diffusion terms
COF2(NN)	Coefficient storage for cylindrical-polar diffusion terms
COF3(NN)	Coefficient storage for cylindrical-polar diffusion terms
CONSRV	Sentinel for conservative or nonconservative differencing of coordinate transformations
CPREF	Reference specific heat
CPTIME	CPU time
CRHS(NCPLD,NEXL*NN)	Right hand side (n^{th} level) of governing equations
CSOLN(NCPLD,NEXL*NN)	Solution array for tridiagonal matrix inversion
CTWO	Coefficient of dilutation stress terms
CVIS1	First constant in Sutherland's viscosity law
CVIS2	Second constant in Sutherland's viscosity law
CVK	von Karman's constant
C1F(NN)	Storage for first coefficient of dissipation equation
C2F(NN)	Storage for second coefficient of dissipation equation

FORTRAN VARIABLE	DESCRIPTION
D	Array index for divergence (integer)
D1(NVSTOR,NPTD,3*NN)	Storage for spatial term coefficients
DELDT	Percentage change for time step adjustment
DELTEW(NN,NN)	Boundary layer thickness
DENSR	Reference density
DFWI(3,3)	Difference weights
DIM1	1.0/Re
DIM2	PREF/(DENSR*WREF**2)
DIM3	PREF/(DENSR*HREF)
DIM4	$(RePr)^{-1}$
DIM5	WREF**2/HREF
DIM6	WREF**2/(Re*HREF)
DIM12	2.0/Re
DISTN(NX,NYZMAX)	Distance to nearest wall
DJDT(NN)	Time rate of change of Jacobian
DJDTAU(NX)	Time rate of change of Jacobian
DLTA(NN)	Boundary layer thickness
DM1(NCPLD,NCPLD,NEXLAM)	Temporary storage for matrix inversion
DM2(NCPLD,NEXLAM)	Temporary storage for matrix inversion
DM3(NCPLD,NCPLD,NEXLAM)	Temporary storage for matrix inversion
DMB(NCPLD)	Temporary storage for matrix inversion
DPHIMX(NESQ)	Maximum time increments of dependent variables
DS	Array index for dissipation (integer)
DSAV1(NEQS,NEQS,3)	Storage for splitting error check
DSAV2(NEQS,NEQS,3)	Storage for splitting error check

FORTRAN VARIABLE	DESCRIPTION
DSAV3(NEQS,NEQS,3)	Storage for splitting error check
DT	Time Step
DTCON	1.0/DT
DTFAC(NN)	Storage for time step
DTMAX	Maximum time step
DTMIN	Minimum time step
DX(NX)	Grid spacing with respect to y^1 direction
DXDY(3,4,2,NX)	Storage used for calculation of Jacobian elements)
DY(NY)	Grid spacing with respect to y^2 direction
DYDX(3,4,2,NX)	Storage used for calculation of Jacobian elements
DZ(NZ)	Grid spacing with respect to y^3 direction
EPSJ1(NN)	Storage used for artificial dissipation
EPSJ2(NN)	Storage used for artificial dissipation
EPSJ3(NN)	Storage used for artificial dissipation
ESTIME	Estimate run time (CPU seconds)
ETAP(NCLMX,3)	Location of cluster points
FORTH(NN)	Storage used for orthogonality check
GAMMA	Ratio of specific heats
GRID(6)	Grid distribution parameters for hyperbolic tangent transformation
GSUP(3,3)	Transformation derivatives for nonorthogonal boundary conditions
H	Array index for enthalpy (integer)
HJCBSV	Saved Jacobian

FORTRAN VARIABLE	DESCRIPTION
HREF	CPREF*TREF
HTOT	Reference value of total enthalpy
I1	Index for first grid point along a grid line
IA	1
IADI	ADI sweep direction (1, 2, or 3)
IB	2
IBCENT	Sentinel for boundary or interior point
IBC	Counter for boundary
IBOUND(NSURMX,3)	Types boundary
IC	3
ICFLAG	Flag for corners
ICONS	Sentinel for convection terms in cylindrical-polar coordinates
ICOORD	Sentinel for coordinate system
IDISTN(NX,NYZMAX)	Index for distance to nearest wall
IDMPX	y^1 grid point for matrix debug dump
IDMPY	y^2 grid point for matrix debug dump
IDMPZ	y^3 grid point for matrix debug dump
IDT	Current time step number
IDTADJ	Sentinel for adjusting time step
IDTGCP	Sentinel for printing of geometry
IDTINT	Initial time step for matrix debug dump
IDTS	Initial time step for number for restart
IDUMP1	Sentinel for printing initial conditions
IEQ	Equation number

FORTRAN VARIABLE	DESCRIPTION
IEQBC(NSURMX,3,NEQS)	Boundary condition input array
IEQNUM(NEQS)	Equation number
IEQSET	Flag for direct input of equations to be solved
IFIRST(25)	Initiators for READMS
IFSB(NX,NZ)	Grid array indicator
IGDMP	Sentinel for matrix debug dump
IGEOM(3)	Sentinel to indicate type of geometry construction
IGPRT(NGVMAX)	Print index for geometry-absolute
IGPRTR(NGVMAX)	Print index for geometry-relative
IGSHF1	Shift for storage of Jacobian elements
IGSHF2	Shift for storage of radii
IGSHF3	Shift for storage of Jacobians
IGSHF4	Shift for storage of coordinates
IGTAPE	Unit for geometry input
IHSTAG	Index for form of energy equation
IJCBN(3)	Index for Jacobians
IJCBNI	Index for inverse Jacobian
IL	Index for last grid point along a grid line
IMACH	Index for Mach number conditioning
INC(3,3)	Array for address increments
INCADI	Number of ADI sweeps
INCOMP	Sentinel for incompressible option
INCGEO	Incremental index for geometry directions
INFILE1	Unit for NAMELIST restart file

FORTRAN VARIABLE	DESCRIPTION
INFILE2	Unit for dependent variable restart file
INITNDX1	Address for boundary conditions
INITNDXB	Address for boundary point
INITNDXG	Address for first interior point
INOUT	Index for in-core or out-of-core
IORTHO	Index for orthogonality
IOFILE1	Unit for NAMELIST output
IOFILE2	Unit for dependent variable output
IPA	Group number of equations presently being solved
IPERBC(3)	Periodic sentinel
IPEROD	Periodic sentinel
IPLOT	Plot interval
IPLOTF	Plot on/off switch
IPRINT	Print interval
IRESTF	Restart on/off switch
IRAD(3)	Index for radii
IRADI	Index for inverse radius
ISMOOTH	Sentinel for smoothing of turbulent viscosity
ISETKE	Time step when k-ϵ calculation initiates
ISSCNT	Sentinel for convergence history print
ISURF(6)	Surface number
ISWIRL	Sentinel for solution of swirl momentum equation
ITCALC	Index for time step conditioning

FORTRAN VARIABLE	DESCRIPTION
ITDGCP	Interval for geometry print
ITPLT	Number of time steps for summary time step print
ITRD(3,4)	Array for indices of Jacobian element storage
IUNITS	Sentinel for metric or English units
IVARNO(NEQS)	Variable associates with equation
IVARPR(60)	Index for dependent and derived variable print
IVISC	Index for viscosity model
IWRGEO	Sentinel for geometry storage on restart
IX1	Radius index for diffusion terms
IX2	Radius index for diffusion terms
IXU1(NX*NZ+1)	Record identifier
IXYZ(3,3)	Coordinate index
JB1	First boundary point index
JB2	Last boundary point index
JEQN(NEQS,3)	Equation number
JGRID	Grid Option
JG1	First interior point index
JG2	Last interior point index
JGEND(NLOOPMAX,NZ)	Index for y^3 derivatives
JGSTART(NLOOPMAX,NZ)	Index for y^3 derivatives
JIND(3)	Index for Jacobian elements
JLR(3)	Index for boundary points
JLRSAV(NLOOPMAX,NZ)	Index for y^3 derivatives
JSEC	Section number for boundary condition

FORTRAN VARIABLE	DESCRIPTION
JVAR(NEQS,3)	Variable number
JX	y^1 index
KTAPE	Unit for plot file
KZ	y^3 index
LA	First matrix element
LB	Second matrix element
LBRKX(NBRKX)	y^3 grid point where boundary condition changes
LBRKZ(NBRKZ)	y^1 grid point where boundary condition changes
LC	Third matrix element
LCORNR	Number of reentrant corners
LCORNX(NCRMAX)	y^1 location of corner
LCORNZ(NCRMAX)	y^3 location of corner
LEQ1	First equation number of set
LEQ2	Last equation number of set
LEV1	Time level n
LEV2	Time level n+1
LEV3	Time level $n+\beta$
LEXE(2,NEXTX,NZ)	y^1 extents for calculation of derived variables
LEXTX(2,NEXTX,NZ)	Y^1 extents for ADI sweeps
LEXTXA(2,NZ)	Absolute y^1 extents
LEXTZ(2,NEXTZ,NZ)	y^3 extents for ADI sweeps
LEXTZA(2,NX)	Absolute y^3 extents
LEZE(2,NEXTZ,NX)	y^3 extents for calculation of derived variables

FORTRAN VARIABLE	DESCRIPTION
LGEDGE(NN,2)	Gridpoint number for edge of boundary layer
LINPR	Sentinel for linearized pressure and temperature
LLCRN	Corner number
LOOP	Loop number
LSEC	Section number for boundary conditions
LSHAPE	Shape option for computational domain
LX	y^1 index
LXRES(NEQS)	y^1 index for maximum residual
LXSST(NEQS)	y^1 index for maximum time increment
LY	y^2 index
LYRES(NEQS)	y^2 index for maximum residual
LYSST(NEQS)	y^2 index for maximum time increment
LZ	y^3 index
LZPRINT(NZ)	y^3 print option index
LZRES(NEQS)	y^3 index for maximum residual
LZSST(NEQS)	y^3 index for maximum time increment
MAXX	Maximum y^1 index
MAXXM1	MAXX-1
MAXY	Maximum y^2 index
MAXYM1	MAXY-1
MAXZ	Maximum y^3 index
MAXZM1	MAXZ-1
MINX	Minimum y^1 index
MINXP1	MINX + 1
MINY	Minimum y^2 index

FORTRAN VARIABLE	DESCRIPTION
MINYP1	MINY + 1
MINZ	Minimum y^3 index
MINZP1	MINZ + 1
MEQK	Last equation solved
MEQS1	First equation
MEQS2	Last equation
MGD1	First interior grid point
MGD2	Last interior grid point
MGDUMP	Option for matrix dumps
MGPRNT	Print interval for matrix dumps
MINF	Free stream Mach number
MINP	Unit for general input
MOVEC	Option for moving coordinates
MPRT	Unit for general output
MSECRF	Reference section number
MSECX	Total y^1 sections
MSECZ	Total y^3 sections
MUINF	Free stream viscosity
NCLUST(3)	Number of cluster points
NCNTR	Center of inline planes or lines
NCOL	Number of columns in matrix
NCUP	Number of coupled equations in a set
NEQN(NEQS,3)	Equation number
NGDIM1	Index for Jacobian elements
NGDIM2	Index for Jacobian elements
NGDIM3	Number of spatial coordinate direction

FORTRAN VARIABLE	DESCRIPTION
NGNYIN	Number of grid points in a plane
NGPTOT1	Set equal to parameter NGPTOT
NGVAR	Number of Jacobian elements
NGVRJ	Number of radii and Jacobian variables
NGVTOT	Total number of geometric variables
NIN	Number of planes in core in 3-D
NJLR3(NZ)	Number of sections for y^3 derivatives
NLIN1(NSEGMAX,NZ)	Number of lines to be simultaneously inverted with respect to the y^1 direction
NLIN2(NSEGMAX,NZ)	Number of lines to be simultaneously inverted with respect to the y^2 direction
NLIN3(NSEGMAX,NZ)	Number of lines to be simultaneously inverted with respect to the y^3 direction
NLOOPS	Number of loops or separate ADI segments
NLOOPX(NZ)	Number of loops with respect to the y^1 direction
NLOOPZ(NX)	Number of loops with respect to the y^3 direction
NLWR(10)	Sentinel for NAMELIST print
NLXE(NZ)	Number of loops with respect to the y^1 direction for updating the derived variables
NLZE(NX)	Number of loops with respect to the y^3 direction for updating the derived variables
NPADI	Number of groups of equations to be solved
NPRES1	Time step number at which pressure on a surface is PRES1

FORTRAN VARIABLE	DESCRIPTION
NPRES2	Time step number at which pressure on a surface is PRES2
NROW	Number of rows of a matrix
NSEG1(NZ)	Number of segments for multiple line inversion with respect to direction y^1
NSEG2(NZ)	Number of segments for multiple line inversion with respect to direction y^2
NSEG3(NZ)	Number of segments for multiple line inversion with respect to direction y^3
NSLAB	Number of slabs of data
NSURF(6)	Number of surfaces
NT	Number of time steps to be run
NTGRID	Total number of grid points
NTREST	Restart interval
NTSTEP	Number of time steps in a cycle
NTWALL1	Time step number at which wall temperature is TWALL1
NTWALL2	Time step number at which wall temperature is TWALL2
NUMBV	Number of endcaps
NUMDX	Number of interior points with respect to direction y^1
NUMDY	Number of interior points with respect to direction y^2
NUMDZ	Number of interior points with respect to direction y^3
NUNERR	Unit for checking accuracy of matrix inversion
NUNIT	Unit for scratch disk
NVDEL	Shift for storage of changes in dependent variables

FORTRAN VARIABLE	DESCRIPTION
NVDEP	Number of dependent variables
NVROT1	Time step number at which rotational speed is VROT1
NVROT2	Time step number at which rotational speed is VROT2
NVTOT	NGVTOT + NVZERO; number of fluid dynamic and geometric variables
NVZERO	NVDEL + NVDEP; number of fluid dynamic variables
NXNYIN	NXIN*NYIN
NXIN	Maximum number of grid points with respect to direction y^1
NX1	First grid points in y^1 direction
NX1C	NX1 + NCNTR - 1
NX1M1	NX1 - 1
NX1P1	NX1 + 1
NX2	Last grid point in y^1 direction
NX2C	NX2 - NCNTR + 1
NX2M1	NX2 - 1
NX2P1	NX2 + 1
NYIN	Maximum number of grid points in y^2 direction
NY1	First grid point in y^2 direction
NY1M1	NY1 - 1
NY1P1	NY1 + 1
NY2	Last grid point in y^2 direction
NY2M1	NY2 - 1
NY2P1	NY2 + 1
NZ1	First grid point in y^3 direction

FORTRAN VARIABLE	DESCRIPTION
NZ1C	NZ1 + NCNTR - 1
NZ1M1	NZ1 - 1
NZ1P1	NZ1 + 1
NZ2	Last grid point in y^3 direction
NZ2C	NZ2 - NCNTR + 1
NZ2P1	NZ2 + 1
NZNYIN	NZIN*NYIN
P	Array index for pressure (integer)
PAMB(NSURMX,3)	Pressure on boundary
PCNT1	Criterion for increasing time step
PCNT2	Criterion for decreasing time step
PI	π
PINF	Free stream pressure
PNOM	Nominal pressure
PREF	Reference pressure
PRES1	Boundary pressure at time step NPRES1
PRES2	Boundary pressure at time step NPRES2
PRESS1(NXZMAX,NYZMXP,6)	Storage for surface static or stagnation pressures
PRLAM	Laminar Prandtl number
PRRAT	Ratio of laminar to turbulent Prandtl numbers
PRTURB	Turbulent Prandtl number
PSTAT1	Pressure for calculation of reference stagnation pressure, PTOT
PTOT	Reference stagnation pressure
PZERO	Free stream stagnation pressure

FORTRAN VARIABLE	DESCRIPTION
R	Array index for density (integer)
REPL	Reynolds number per unit length
RESAVG(NEQS)	Average residuals
RESEQ(NEQS)	Maximum residuals
REY	Reynolds number
RGAS	Gas constant
RHO1	Density for calculation of reference stagnation pressure, PTOT
RHOINF	Free stream density
SAVE(NEQS,NEXL*NN)	Storage for changes of dependent variables
SCLAM	Laminar Schmidt number
SCTURB	Turbulent Schmidt number
SCRAT	Ratio of laminar to turbulent Schmidt numbers
SLOPE(NCLMX,3)	Specified changes for method of Oh
SOUND	Sound speed
SSEPS	Criterion for convergence
SSTEST	Maximum time increment of dependent variables
T	Array index for temperature (integer)
TAUW	Wall Shear
TDUM(NN)	Temporary storage array
TDUMRES(NN)	Storage for residuals
TED	Array index for dissipation (integer)
TEDMIN	Minimum dissipation
TINF	Free stream temperature

FORTRAN VARIABLE	DESCRIPTION
TKE	Array index for turbulence kinetic energy (integer)
TKEMIN	Minimum turbulence kinetic energy
TLEFT	Time left (CPU seconds)
TMLC	Array index for mixing length
TREF	Reference temperature
TSUM(NN)	Temporary storage array
TTIME	Elapsed Time
TTOT	Reference total temperature
TVAR(NTPLT,NPLVAR)	Time variable storage
TWALL(NSURMX,3)	Temperature on boundary
TWALL1	Boundary temperature at time step NTWALL1
TWALL2	Boundary temperature at time step NTWALL2
TWOD	2-D or 3-D sentinel
TZERO	Free stream stagnation temperature
U	Array index of y^1 velocity component (integer)
UINF	Free stream velocity
V	Array index for y^2 velocity component (integer)
VELMX(NN)	Maximum velocity
VISCIN	Input viscosity
VISCR	Reference viscosity
VROT1	Rotation speed at time step NVROT1
VROT2	Rotation speed at time step NVROT2
VS1	Array index for laminar (molecular) viscosity (integer)

FORTRAN VARIABLE	DESCRIPTION
VS2	Array index for effective viscosity (integer)
W	Array index for y^3 velocity component (integer)
WMINBC	Minimum velocity for certain inflow boundary conditions in subroutine BC
WREF	Reference velocity
WTREF	Reference molecular weight
X(NX)	Storage for x coordinate
XCORN(NCRMAX)	x locations of corners
XDIM2(NN)	Temporary storage
XDIM25(NN)	Temporary storage
XDIM3(NN)	Temporary storage
XLOC(NX)	Storage for x coordinate
XMIN	Minimum value of x
XMAX	Maximum value of x
Y(NY)	Storage for y coordinate
YLOC(NY)	Storage for y coordinate
YMIN	Minimum value of y
YMAX	Maximum value of y
Z(NZ)	Storage for z coordinate
ZCORN(NCRMAX)	z locations of corners
ZLOC(NZ)	Storage for z coordinate
ZMIN	Minimum value of z
ZMAX	Maximum value of z
ZN	Array index for inverse of molecular weight (integer)
ZREF	Inverse of reference molecular weight

5.0 LOGICAL FILE UNITS UTILIZED BY THE CMINT COMPUTER CODE

The CMINT computer code utilizes up to ten (10) logical file units during the execution of a run. In many cases not all ten units are used, and hence, in these cases it is not necessary to define all ten units. All references to logical file units in the CMINT computer code are accomplished through the use of a FORTRAN name rather than through a specific unit number. Thus, if the user desires to change a logical file unit number, this can be accomplished easily. A list of the logical file units utilized by the CMINT computer code, their FORTRAN names, the default value unit number and a brief description of the use of the unit is presented below.

FORTRAN Name	Default Unit Number	Description
KTAPE	1	Plot file.
NUNERR	2	Scratch file for matrix inversion error check.
MINP	5	Input file for initial NAMELIST input.
MPRT	6	Printed output unit.
INFILE1	9	Input unit number for NAMELIST restart data.
IOFILE1	10	Output unit number for NAMELIST restart data.
NUNIT	11	Scratch file for out-of-core option.
INFILE2	19	Input unit number for dependent variable restart data.
IOFILE2	20	Output unit number for dependent variable restart array.
IGTAPE	21	Input unit for geometric data.
-----	51	JANNAF thermochemical binary data file. (Created using LeRC CET86 code.)

FORTRAN Name	Default Unit Number	Description
-----	52	File for chemical equilibrium curve fit or table look-up binary data. Created using modified CET86 code.

6.0 STRATEGY FOR RUNNING THE CMINT COMPUTER CODE

The following discussion is meant to give the user a basic understanding of the strategy for successfully running the CMINT computer code. It is suggested that the user read and understand this section before attempting to prepare input for the CMINT code. Basically, the strategy for running the CMINT code can be divided into six areas: (1) specification of the system of units to be employed, and reference flow conditions, (2) generation of the coordinate system in which the calculation is to be performed, (3) specification of the boundary conditions, (4) generation of the initial conditions, (5) selection of the time step to insure rapid convergence of the solution for steady state problems or for physically realistic transients for time accuracy, and (6) solution of the governing equations with a k-ϵ turbulence model.

Reference Flow Conditions

Preparation of the input for the CMINT computer code first requires the choice of a system of units to be used for the nondimensionalization of the governing equations (all variables are nondimensionalized with respect to the freestream conditions). Two options are available in the CMINT code, viz., the SI system and the English system, corresponding to values of the FORTRAN variable IUNITS = 1 and 2, respectively. Table 2 shows the various reference variables, and the units in both the SI and English systems. Reference condition input for the CMINT code can be divided into three categories which are referred to as: (1) flight conditions, (2) wind tunnel conditions, and (3) primitive conditions and refer to the types of flow condition information that would normally be available to the user. Table 3 shows the required input for each option. Adequate discussion of the types of reference conditions requires the use of several simple relationships which relate the various flow conditions. For a single species perfect gas the relations are given below; and for a multiple species mixture of gases the relations are given in Appendix A. The perfect gas law is

$$P_\infty = \rho_\infty R T_\infty \tag{1}$$

and the definition of Mach number is

$$M_\infty = \frac{U_\infty}{\sqrt{\gamma R T_\infty}} \tag{2}$$

The isentropic relationships along a streamline for a perfect gas with constant specific heat, c_{p_∞}, are

$$\frac{P_0}{P_\infty} = \left[1 + \frac{\gamma-1}{2} M^2_\infty\right]^{\gamma/(\gamma-1)} \tag{3}$$

$$\frac{T_0}{T_\infty} = 1 + \frac{\gamma-1}{2} M^2_\infty$$

The definition of Reynolds number per unit length is

$$\frac{Re}{\ell_\infty} = \frac{\rho_\infty U_\infty}{\mu_\infty (T_\infty)} \tag{5}$$

Sutherland's viscosity law, which assumes that the viscosity of a thermally perfect gas is a function of temperature (T_∞) alone,

$$\mu_\infty (T_\infty) = \frac{c_1 T_\infty^{3/2}}{c_2 + T_\infty} \tag{6}$$

The specific heat relationship for a perfect gas is

$$c_{p_\infty} = \frac{\gamma}{\gamma-1} R \tag{7}$$

and the definition of enthalpy for a calorically heat perfect gas is

$$h_\infty = c_{p_\infty} T_\infty \tag{8}$$

When flight conditions are given, typically a reference length, ℓ_∞, freestream Mach number, M_∞, static temperature, T_∞, and static pressure, P_∞, are specified. Given these variables the freestream or reference velocity, U_∞, can be calculated from Eq. (2), while Eq. (1) can be used to calculate the freestream density ρ_∞, and Eq. (5) can be used to calculate the Reynolds number per unit length. Eqs. (3) and (4) yield the freestream stagnation pressure and temperature while Eqs. (7) and (8) yield the freestream specific heat and enthalpy.

When wind tunnel conditions are given, the normal input information usually consists of the reference length, ℓ_∞, Mach number, M_∞, the Reynolds number per unit length, Re/ℓ_∞, and either the static pressure, P_∞, or the stagnation pressure, P_0. By combining Eqs. (1), (2), (5) and (6) a quadratic equation for T_∞ can be obtained in terms of known variables, i.e.,

$$T_\infty = \frac{B + \sqrt{B^2 + 4c_1c_2B}}{2c_1} \tag{9}$$

where

$$B = \frac{\sqrt{\frac{\gamma}{R}}\, M_\infty P_\infty}{\frac{Re}{\ell_\infty}} = \frac{\sqrt{\frac{\gamma}{R}}\, M_\infty P_0}{\frac{Re}{\ell_\infty}\left[1 + \frac{\gamma-1}{2} M^2\right]^{\gamma/(\gamma-1)}} \tag{10}$$

Eqs. (1), (2), (4), (6), (7) and (8) can then be used to calculate freestream density, velocity, stagnation temperature, viscosity, specific heat and enthalpy, respectively.

With the so-called primitive option, the input consists of the reference length, ℓ_∞, the freestream velocity, u_∞, freestream pressure p_∞, and the freestream temperature, T_∞. Eqs. (1) - (8) can then be used to calculate the remaining freestream values.

Coordinate System

The CMINT computer code has a very general coordinate capability. Calculations can be performed in either a Cartesian (ICOORD = 0) or a cylindrical-polar (ICOORD = 1) reference system in either two or three dimensions. All computations ultimately are performed in computational space denoted by y^1, y^2 and y^3. For the Cartesian systems the coordinates are x, y and z (x_1, x_2 and x_3), while in cylindrical-polar systems the coordinates are r, θ and z. All velocity components, as well as the unit vectors for the momentum equations, are in the appropriate untransformed reference system. However, all derivatives are fully transformed, giving a general nonorthogonal coordinate capability. The user must supply the appropriate coordinate information which consists of matched grid point

and Cartesian locations. All transformation derivatives are automatically calculated in subroutine TIMGEO.

Coordinate information can either be calculated externally to the CMINT code and input to the code via file unit IGTAPE, calculated within the CMINT code or some combination of the two. Control of the coordinate information is determined by the input variable IGEOM(IDIR) where IDIR refers to the directions in computational space. A value of IGEOM(IDIR) equal to 1 refers to data that is externally produced and read into the CMINT code from unit IGTAPE. A value of IGEOM(IDIR) = 2 refers to data generated within the CMINT code by use of subroutine TANHYP and value of IGEOM(IDIR) = 3 refers to data generated by subroutine OHGRID. Basically subroutines TANHYP and OHGRID are used to distribute grid points on the boundaries of rectangular physical domains by the use of a hyperbolic relationship between physical and computational space or by the method of Oh (Ref. 6) which uses error functions for the same purpose. The hyperbolic tangent method is useful when the user wants to pack grid points near boundaries. This is done by the selection of the packing parameter TTI(IBC) appropriate to the given surface. Values of TTI vary from 0 to 1 with 0 corresponding to equal spacing. As values approach 1 the packing rate increases at the boundary. The method of Oh is considerably more general than the TANHYP method and is discussed in detail in Section 7. Input variables required for the method of Oh are CLPX, CLPY, NCLUST, ETAP, ALPHA and SLOPE which are discussed in the input section.

When coordinate data is read into the CMINT code it must naturally be in a certain prescribed form. The CMINT convention utilizes binary read statements. Each record must contain a y^1-y^2 computational plane of the coordinate variable specified by the IGEOM(IDIR) = 1 option and there must be one record for each of the NZIN y^3's. For two-dimensional cases, y^2 has a dimension of 1; hence each record is of length of y^1, i.e., NXIN. If two or more of the IGEOM(IDIR)'S are equal to 1, then the record for the first variable must be followed by the record for the second variable, etc. until all the y^1-y^2 planes of data have been input. When IGEOM(3) $\neq$ 1 and IGEOM(1) = IGEOM(2) = 1, then it is possible to 'stack' data. In this case, file IGTAPE needs to contain only two records. The first is the values of x or r on the first y^1-y^2 plane and the second is the values of y or θ on the first y^1-y^2 plane. Since these values are assumed constant on subsequent, i.e., later values of y^3 planes, the input unit is rewound at each y^3 station and the data read in again. This is particularly useful if planes of

coordinate data are parallel. For the details of the coordinate generation process in the CMINT code the user is referred to the listing of the code. All coordinate and transformation data is generated within subroutine TIMGEO. Coordinate generation is within the 300 DO loop.

Specification of Boundary Conditions

A wide variety of boundary conditions are available as input to the user of the CMINT computer code. In addition, there are default values of boundary conditions which can be activated by identifying the boundaries with the FORTRAN variable IBOUND(KSURF,IDIR) where KSURF is the surface number with respect to the IDIRth direction (for discussion of convention used for KSURF and IDIR see section on Protocol for Grid Point and Boundary Identification). Values of IBOUND = 1, 2, 3 or 4 correspond to an inlet, a symmetry surface, an exit plane or a wall, respectively. Default values of the boundary conditions (FORTRAN variable (IEQBC(KSURF,IDIR,IEQ)) associated with specific values of IBOUND are described in the NAMELIST input description section. Any default value of IEQBC can be overwritten by inputting a value of IEQBC in the NAMELIST input.

The default boundary conditions are contained within subroutine BCSET. Basically the procedure requires that IBOUND(KSURF,IDIR) be specified. For inlet (IBOUND = 1) the default boundary conditions are that the total pressure condition, PTOT, be associated with the streamwise momentum equation, that the cross flow velocity components are zero, that the second derivative of pressure equals zero, that the stagnation enthalpy equals the reference total enthalpy, HTOT and that the values of k and ϵ correspond to the equilibrium mixing length values. For symmetry surfaces (IBOUND = 2) symmetry conditions are used, i.e., the normal velocity component is zero while first derivatives of the remaining velocity components, pressure, temperature and k and ϵ are zero. For exit planes (IBOUND = 3) linear extrapolation (second derivative = 0.0) is used for all variables except pressure, which is specified by PAMB. For walls (IBOUND = 4) the no-slip conditions are assumed for the velocity components. The pressure condition used for the continuity equation boundary condition is that the first derivative of pressure is zero (the boundary layer condition). The thermal condition assumes an adiabatic wall unless a wall temperature, TWALL, is input. The turbulence kinetic energy, k, and the turbulence energy dissipation rate, ϵ, are assumed to be zero on the wall.

One boundary condition that needs some explanation is the so-called 'two-layer model' which is used as one of the default boundary conditions on an inlet. This boundary condition assumes that the flow in an inlet is divided into two regions or layers; i.e., a central inviscid core where the stagnation pressure is constant, and attached boundary layer(s) where the pressure is constant and the velocity profile has a prescribed form. The prescribed normalized form (but not the magnitude) of the boundary layer velocity profile must be supplied by the user. The CMINT code presently allows the boundary layer profile(s) to be input via namelist $READ5 or it may be coded directly in the initial profile subroutines SPREAD, SPREADTP or SPREAD2. See the following section, Generation of Initial Conditions. One feature of the two-layer model is that the mass flux will be determined from the solution of the governing equations, and not from an imposition of the inlet profiles of velocity and density and will be consistent with the imposed total pressure ratio P_0/P_{exit}.

Generation of Initial Conditions

Since the steady state solution is obtained by time marching the governing equations until the solution ceases changing, an initial guess of all flow variables must be generated. Because of the wide variety of geometric configurations considered, it would be impossible to develop a single technique for obtaining initial conditions. Rather, what is done is to default all initial conditions to the stagnation conditions (see subroutine FLWFLD) or to allow the user to supply subroutine SPREAD, which can be adapted to generate a desired consistent set of initial conditions. Also, the user may refer to optional initial profile routines SPREADTP for turbulent pipe flow initialization and SPREAD2 for general external flow initialization (i.e., flows with no inlet boundary layers). The initial profile routine is selected by specification of the flag ISPREAD = 1, 2 or 3 for routines SPREAD, SPREADTP or SPREAD2, respectively. The default stagnation initial condition has proven to be useful for a wide variety of problems. All conditions are set to their stagnation values and the back pressure is lowered over a series of time steps through the use of the IEQBC = 5 boundary condition for the continuity equation at the downstream boundary. This causes the flow to drain through a device (in the case of internal flows) and a flow field established and then converged, once the back pressure reaches its preset value of PRES2. For external flows it is often reasonable to

set freestream values everywhere (in subroutine SPREAD or SPREAD2) and to append constant pressure boundary layers on solid surfaces as an initial condition and let these conditions time march to convergence.

Selection of Time Step

The selection of the time step, DT, is critical to obtaining a converged solution for steady state cases, while for time dependent cases it is also critical in obtaining an accurate transient analysis. In time dependent cases the selection of time step is greatly dependent upon the case. The main criterion is that the time step be small enough to resolve the transient phenomena associated with the case. The basic procedure is to choose a representative time step and to monitor the changes. If the changes are too large, the time step must be reduced. For steady state calculations, several options are available. Since, in this case, the steady state solution is the objective, any means of obtaining that end are acceptable. The basic procedure employed in the CMINT computer code is to use a spatially varying time step to accelerate convergence. This is done by the ITCALC input variable. Acceptable values of ITCALC are 0, 1, 2, 3, 4 or 5. A value of ITCALC equal to 0 implies a constant (non-spatially varying) time step. If the value of ITCALC is odd, the so-called Mach number scaling technique is employed on the time density term to under-relax the density. This technique is especially useful for low Mach number cases but should not be used for a Mach number of 0.3 or greater, as a general rule. Additional details concerning the selection of time steps can be obtained in the section on NAMELIST input under section $READ4. It has been found that if a steady state solution is desired, ITCALC values of 2, 3, 4 or 5 are usually a more efficient means of obtaining such a solution. Time steps on the order of 0.5 through 10.0 are usually good choices for DT. Usually, some experimentation must be done to obtain the optimum. Often a reasonable approach is to use the IDTADJ = 1 option and to let the code choose the value of DT, which will vary between DTMIN and DTMAX. Usually, the basic flow field should develop within 100 time steps and a converged solution may be obtained within 300 to 500 time steps. In some complicated flow problems, several thousand time steps may be required to obtain a converged solution. Obviously, the above numbers are approximate, as the convergence rate is case dependent. However, if the solution is only slowly converging, this is usually a sign that the time step is too small and the user may want to select a larger time step.

Other problems affecting convergence are poor grid resolution in critical regions and improper boundary condition specification.

The k-ϵ Turbulence Model

Solution of the governing equations with a k-ϵ turbulence model requires the solution of two additional partial differential equations. The procedure used in the CMINT computer code is to first solve the flow equations of conservation of mass, momentum and energy with a mixing length model to develop the basic flow patterns and then to switch to a k-ϵ turbulence model. In some cases, it is advantageous to initially freeze the fluid dynamics and then to solve the k-ϵ equations in isolation to obtain the basic k-ϵ fields and then to solve the fluid dynamics and k-ϵ equations simultaneously until a converged solution is obtained. All of the above processes are controlled by the FORTRAN input variables IVISC and ISETKE. Initially, IVISC is set equal to 3, the mixing length model. At time step ISETKE, the code will automatically take the flow field and mixing length, assume that production equals dissipation in the k equation and produce a consistent k-ϵ initial guess. With this initial condition, both fluid dynamics and k-ϵ fields, will then be time iterated to convergence. If it is desired to solve the k-ϵ equations with frozen fluid dynamics, the value of ISETKE should be set equal to the negative of the desired time step.

7.0 TRANSFORMATION FUNCTION

The distribution of grid points on boundaries is accomplished by the use of a transformation technique developed by Oh and described in Ref. 6. If y and η designate the independent variables in physical and computational (grid point) space, respectively, the transformation function

$$f = \frac{dy}{d\eta} \tag{11}$$

can be integrated to yield

$$y = \int_{\eta_{min}}^{\eta} f(\eta)d\eta + y_{min} \tag{12}$$

A convenient transformation function is composed of a series of N complementary error functions of the form

$$f(\eta) = \frac{dy}{d\eta} = \beta_0 + \frac{1}{2} \sum_{j=1}^{N} \left\{ \operatorname{erfc}\left[\frac{1}{\alpha_j}\left(\eta - \eta_{p_j}\right)\right] - \left[1 + \operatorname{sign}(\alpha_j)\right] \right\} \beta_j \tag{13}$$

The j^{th} complementary error function is centered in computational space at location η_{p_j} (which is referred to as a pivot point), and α_j is the width in computational space in which 90% of the grid size variation takes place. At computational space location η_{p_j} the values of $f(\eta)$ will assume a local maximum. In the limit as $\alpha_j \to 0$, β_j is the j^{th} step height for pivot j, i.e., the difference in grid spacing on either side of η_{p_j}.

Substitution of Eq. (13) into Eq. (12) and integration yields

$$y - y_{min} = \beta_0(\eta - \eta_{min}) + \sum_{j=1}^{N} \frac{1}{2} \left\{ a_j \left[\theta_j(\eta) = -\theta(\eta_{min})\right] - \left[1 + \operatorname{sign}(\alpha_j)\right]\left[\eta - \eta_{min}\right] \right\} \beta_j \tag{14}$$

where

$$\theta_j(\eta) = \frac{1}{\alpha_j}\left(\eta - \eta_{\rho_j}\right)\ \mathrm{erfc}\left[\frac{1}{\alpha_j}\left(\eta - \eta_{p_j}\right)\right] - \frac{1}{\sqrt{\pi}}\, e^{-\left[1/\alpha_j\left(\eta - \eta_{p_j}\right)\right]^2} \tag{15}$$

The technique used in this study is to constrain the values of the physical coordinate, y_{c_k}, at specific values of the computational coordinate, η_{c_k}. At interior points the η_{c_k}'s are referred to as interior cluster points. At the two end points, there exist pairs of computational and physical points η_{min}, y_{min}, and η_{max}, y_{max} which are referred to as end cluster points. The use of a finite series of formulation enables a set of linear constraint equations to be formulated. In the context of obtaining one-dimensional distribution functions this involves setting constraints on location and possibly grid spacing at a given grid point. Thus for each constraint a unique linear algebraic equation is derived. For this purpose equation 14 is used for location constraint and equation 13 for grid spacing constraint. The end result of repeated application of equations 13 and 14 as appropriate is a set of linear algebraic equations in β_j. Clearly since $1 \leq j \leq N$, N values of η_{pj} and α_j are required. Figure 1 illustrates the rules for choosing these values. In general at a point at which location alone is constrained a single complementary error function is required while if both location and spacing are constrained two are required. When location above is constrained the function can be i- the interval before or after the grid point to be constrained. When both constraints are applied experience has shown that one pivot point on both sides of the grid piont produces the best results. The system of equations is solved for the β's by standard Gaussian elimination techniques and hence all the constants of Eq. (14) are uniquely determined.

There are several advantages to the use of the series of complementary error functions:

(1) $f(\eta)$ is positive, finite and non-zero, i.e., y will always increase with increasing η;

(2) $f(\eta)$ is continuous and successively differentiable and integrable; and

(3) if the pivot points are spaced at a greater distance (in computational space) than $\alpha/2$ from each other, the complementary error functions will have minimal interaction. Thus, the width and location of a

complementary error function can be changed without affecting other complementary error functions.

The strategy for using this technique is as follows:

(1) Determine the location of the interior and end cluster points in physical and computational space. The criteria used in determining these locations is usually based upon the physical processes that need to be resolved and the number of grid points needed to resolve those processes;

(2) The pivot points, η_{p_j}, and the corresponding bandwidth parameters, α_j, are then input. Usually the interior pivot points are located on either side of the cluster points. If it is desired to have negligible interaction between the complementary error functions, the η's and α's will be chosen such that no η will be located within $\alpha/2$ grid points of each other;

(3) Slopes are then chosen at each interior and end cluster point. Again these are usually determined from physical considerations;

(4) A test run is then made with a stand-alone version of SUBROUTINE OHGRID. Usually the initial results will not be precisely to the liking of the user. In general this will be due to the choice of the slope constraints which can then be changed to obtain a better quality grid point distribution; and

(5) Once this interaction process has been completed with the slope constraints (because of item 3 in the advantages of this method that process should be rapid), slight modification of the values of the η's and α's should result in the desired grid point distribution.

Intentionally left blank.

8.0 PROTOCOL FOR BOUNDARY IDENTIFICATION

Because of the complexity of the configurations which may be considered a protocol had to be developed so that the user can easily set up the finite difference mesh required to solve the governing equations and apply the appropriate boundary conditions. Refer to Figure 2 for a schematic of a sample computational domain with embedded solid bodies in the x-z plane. CMINT requires specification of all re-entrant and inward corner points in the domain via the variables LXCORN(ICRN) and LZCORN(ICRN), which must be matched pairs for each corner number ICRN (NAMELIST $READ2). CMINT Version 5.04 does not permit re-entrant corners in the y^2 direction.

It is convenient at this point to explain the surface number (KSURF) convention and direction (IDIR) convention utilized throughout the CMINT computer code. The correct data input for several variables, e.g. IBOUND(KSURF,IDIR), require a knowledge of this convention. The boundary surfaces are numbered according to a surface number relative to the computational direction. The convention is to allow values of IDIR = 1 and 3 to correspond to the y^1 and y^3 computational directions, respectively. Referring to Figure 2, it can be seen that the surfaces (with respect to the y-direction) are numbered 1x to 10x. This corresponds to values of KSURF ranging from 1 to 10 and value of IDIR corresponding to the y^1- direction, i.e., IDIR = 1. The analogous convention is also applied to the y^3- direction surfaces, which are numbered 1z to 10z. Thus, the value of KSURF represents the surface number relative to the given direction, IDIR.

A schematic of the projectile base flow computational domain is shown in Figure 3.

Intentionally left blank.

9.0 CMINT NAMELIST INPUT

NAMELIST $CONTROL — CMINT Options

VARIABLE NAME	DESCRIPTION
MINT	Control flag. MINT = 1: Fluid dynamic calculation only. MINT = 2: Coupled fluid dynamic and solid heat conduction calculation. MINT = 3: Solid heat conduction calculation only. Default value is 1. At present, options MINT = 2 and 3 require code modifications for each specific case.
NTRAT	Ratio of number of time steps in fluids calculation to number in heat conduction calculation. Default value is 1.
NTTOTL	Not used at present.
IPRHEAT	Not used at present.
IHFLAG	Not used at present.

NAMELIST $READ1 — Restart Options

VARIABLE NAME	DESCRIPTION
INFILE1	Input unit number for NAMELIST restart data. Default value is 9.

NAMELIST $READ1 (continued)

VARIABLE NAME	DESCRIPTION
INFILE2	Input unit number for geometric and fluid dynamic arrays. Default value is 19.
IOFILE1	Output unit number for NAMELIST restart data. Default value is 10.
IOFILE2	Output unit number for geometric and fluid dynamic arrays. Default value is 20.
IREST	Restart flag. IREST = 0: New calculation IREST = 1: Calculation run from a restart file. Default value is 0.
ISTRT3D	Sentinel for optional initialization of a 3-D flow field using a 2-D restart file (ISTRT3D = 1). Calls subroutine INIT3D, which may require user modification. Default value is 0.
NFINP	Number of files to be skipped on input devices INFILE1 and INFILE2. Not operational. The first file is read.
NFOUT	Number of files to be skipped on output devices IOFILE1 AND IOFILE2. Not operational. The first file is read.

NAMELIST $READ2 — Geometric and Grid Options

ALPHA(I,IDIR)	Oh transformation parameter. See Section 7. The I^{th} width spacing corresponding to the location of an error function at ETAP(I,IDIR). The width spacing is the number of grid points in which 90% of the grid spacing occurs around the pivot point. 2*NCLUST(IDIR) + 2 values are required, placed as described in Section 7. Input required when IGEOM(IDIR) = 2. No default values.
CLPX(I,IDIR)	Oh transformation parameter. See Section 7. The physical location corresponding to the I^{th} cluster point associated with direction IDIR. The first and last physical location for a direction, IDIR must always be input, e.g., XMIN and XMAX. Input required when IGEOM(IDIR) = 2. No default values.

NAMELIST $READ2 (continued)

VARIABLE NAME	DESCRIPTION
CLPY(I,IDIR)	Oh transformation parameter. See Section 7. The I^{th} grid point number corresponding to the cluster point associated with direction IDIR. The first and last grid points for a direction, IDIR, must always be input. Input required when IGEOM(IDIR) = 2. No default values.
DAMPG(IDIR)	Stretching parameter for the hyperbolic sine grid point distribution in subroutine TANHYP. The value corresponds to the amount of stretching at an interior cluster point at physical location XCENTR(IDIR). A value of DAMPG of 0.0 will yield an equally spaced grid, while a positive value of DAMPG resolves a region of width on the order of 1.0/DAMPG. Default values are 3 * 0.0).
DOUBLVAL	Logical flag which indicates, if .TRUE., that double values for all variables will be utilized at re-entrant corner points (see Figure 2). Default value is .TRUE.
ETAP(I,IDIR)	Oh transformation parameter. See Section 7. The I^{th} grid point location of a pivot point. One value is required for each location constrained grid point and an additional value if both location and grid spacing are constrained. The value of ETAP is the grid point location (may be noninteger) of the center of the error function used in the transformation. The rate of change of grid spacing will be the largest at the pivot points. Input is required when IGEOM(IDIR) = 2. No default values.
GRID(2*IDIR)	Stretching parameters for use with the hyperbolic tangent grid point distribution in SUBROUTINE TANHYP. Two values must be input for each direction corresponding to the amount of stretching at the minimum and maximum extremes. Values of GRID of 0.0 will yield an equally spaced grid, while values of GRID approaching 1.0 will progressively yield a tighter grid at the extremes. Input required when IGEOM(IDIR) = 3. Default values are 6*0.0.

NAMELIST $READ2 (continued)

VARIABLE NAME	DESCRIPTION
ICONS	Sentinel for the cylindrical-polar formulation to determine the differencing for the radial direction derivative convection term. ICONS = 0: Nonconservative differencing with respect to r. ICONS = 1: Conservative differencing with respect to r. Default value is 1.
ICOORD	Sentinel which determines if transformed Cartesian or cylindrical-polar formulation of governing system of partial differential equations is to be utilized. ICOORD = 0: Cartesian formulation ICOORD = 1: Cylindrical-polar formulation Default value is 0.
IDMPX, IDMPY, IDMPZ	Grid point locations for computational direction y, y^2 and y^3, respectively, at which a block-tridiagonal matrix inversion diagnostic print will be output. Valid at interior grid point locations only. A print of matrix coefficients and a check of solution accuracy is output for an entire line for each ADI sweep. See MGDUMP, MGPRNT and IDTINT. Default values are 3 * 0.
IDTINT	Initial time step for matrix inversion diagnostic print. Default value is the first time step of the current run.
IGEOM(IDIR)	Geometric sentinel. IGEOM(IDIR) = 1: Geometry for direction IDIR generated externally and written on unit IGTAPE. Each record consists of a y^1–y^3 plane of data for the $IDIR^{th}$ direction. IGEOM(IDIR) = 2: Geometry generated by the method of Oh. See Section 7. IGEOM(IDIR) = 3: Geometry generated by the use of the hyperbolic tangent. Default values are 3*1.
IGTAPE	Geometry input unit number. Required when any value of IGEOM(IDIR) = 1. Default value is 21.

NAMELIST $READ2 (continued)

VARIABLE NAME	DESCRIPTION
JGRID	Grid type sentinel. JGRID = 0: Normal computational grid. = 1: C-type cascade grid. = 2: O-type cascade grid. = 3: H-type cascade grid. = 4: C-H type rotor-stator grid with moving deformable blending region. Default value is 0.
LBRKX(JSECX)	Grid point break locations for y^1-direction boundary conditions. The break location is the last grid point (in the y^3-direction) at which boundary conditions for section JSECX are used. This flag is set by the code in subroutine CORNER for domains as in Figure 2. If additional break locations other than those associated with re-entrant corners are required, then the entire array must be input.
LBRKZ(JSECZ)	Grid point break locations for y^3-direction boundary conditions. The break location is the last grid point (in the y^1-direction) at which boundary conditions for section JSECZ are used. This flag is set by the code in subroutine CORNER for domains as in Figure 2. If additional break locations other than those associated with re-entrant corners are required, then the entire array must be input.
LSHAPE	Shape of computational space. LSHAPE = 0: Rectangular shape. LSHAPE = 6: Shape of computational domain input through use of corner point locations: LXCORN and LZCORN. Default value is 0.
LXCORN(LCRN)	y^1-direction location of all inward and re-entrant corner points. See Figure 2. Used only when LSHAPE = 6 option is selected. Must be matched with LZCORN(LCRN). No default values.
LXTPLT	y^1-direction grid point location for time history data. Default is 0.
LYTPLT	y^2-direction grid point location for time history data. Default is 0.

NAMELIST $READ2 (continued)

VARIABLE NAME	DESCRIPTION
LZCORN(LCRN)	y^3-direction location of all inward and re-entrant corner points. See Figure 2. Used only when LSHAPE = 6 option is selected. Must be matched with LXCORN(LCRN). No default values.
LZTPLT	y^3-direction grid point location for time history data. Default is 0.
MGDUMP	Matrix inversion print control flag. MGDUMP = 0: No print. = 1: Coefficient print only. = 2: Coefficient print and matrix error check print. = 3: Normalized coefficient print. = 4: Normalized coefficient print and matrix error check print. Default value is 0.
MGPRNT	Time step increment for matrix inversion diagnostic print. Default value is the last time step of the current run.
MOVEC	Sentinel for moving coordinates. MOVEC = 0: Stationary coordinate system. MOVEC = 1: Cascade stator-rotor moving coordinate system. See JGRID flag. MOVEC = 2: Prescribed radial motion of wall. See subroutine SEALG. MOVEC = 3: Reserved for free surface motion problem. Default value is 0.
NCLUST(IDIR)	Oh transformation parameter. See Section 7. Number of internal cluster points needed in SUBROUTINE OHGRID. Input required when IGEOM(IDIR) = 2. No default values.
NIN	Sentinel indicating the number of computational planes kept in memory by the code when using the 3-D out-of-core option, which is used by the code when there is insufficient memory to run the complete grid in core. See the parameter NGPTOT. Default value is 5.
NSLOPE(IDIR)	Oh transformation parameter. See Section 7. Number of slope constraint points needed in subroutine OHGRID. Input required when IGEOM(IDIR) = 2. No default values.

NAMELIST $READ2 (continued)

VARIABLE NAME	DESCRIPTION
NUMDX	Number of interior grid points in the y^1 direction. No default value.
NUMDY	Number of interior grid points in the y^2 direction. No default value.
NUMDZ	Number of interior grid points in the y^3 direction. No default value.
NX1,NX2	Lower and upper grid point range for y^1 direction. Calculated by code.
NY1,NY2	Lower and upper grid point range for y^2 direction. Calculated by code.
NZ1,NZ2	Lower and upper grid point range for y^3 direction. Calculated by code.
SLOPE(I,IDIR)	Oh transformation parameter. See Section 7. The I^{th} spacing constraint in direction IDIR. One value must be input for each value of YSLOPE(I,IDIR). There are NSLOPE values required. No default values.
TWOD	Two-dimensional sentinel. TWOD = .TRUE.: two-dimensional calculation. TWOD = .FALSE.: three-dimensional calculation Default value is .TRUE.
XCENTR(IDIR)	Physical clustering location in IDIR direction for the hyperbolic sine grid transformation in subroutine TANHYP. See stretching parameter description, DAMPG(IDIR). Default values are 3 * 0.5.
XMIN	Dimensionless y^1 minimum distance. Input required when IGEOM(1) = 1 or 2. Default value is 0.0.
XMAX	Dimensionless y^1 maximum distance. Input required when IGEOM(1) = 1 or 2. Default value is 0.0.
YMIN	Dimensionless y^2 minimum distance. Input required when IGEOM(1) = 1 or 2. Default value is 0.0.
YMAX	Dimensionless y^2 maximum distance. Input required when IGEOM(1) = 1 or 2. Default value is 0.0.

NAMELIST $READ2 (continued)

VARIABLE NAME	DESCRIPTION
YSLOPE(I,IDIR)	Oh transformation parameter. See Section 7. The computational coordinate location of the I^{th} spacing constraint in direction IDIR. No default values.
ZMIN	Dimensionless y^3 minimum distance. Input required when IGEOM(1) = 1 or 2. Default value is 0.0.
ZMAX	Dimensionless y^3 maximum distance. Input required when IGEOM(1) = 1 or 2. Default value is 0.0.

NAMELIST $READ3 Flow Conditions and Solution Options

See Table 3 for required reference condition input.

VARIABLE NAME	DESCRIPTION
AFADD	Nondimensional freestream turbulence kinetic energy. Default value is 0.0.
ARTCOM	Logical sentinel for artificial compressibility option (if .TRUE.). Default value is .FALSE.
ARTC1	Artificial compressibility parameter for the continuity equation. Default value is 1.0.
ARTC2	Artificial compressibility parameter for the momentum equations. Default value is 1.0.
AVISC(IDIR,IEQ)	Artificial dissipation for direction IDIR in equation IEQ. The reciprocal of AVISC is the cell Reynolds number. Default values are (3*NEQS)*0.5.
BALDLOMX	Logical flag to activate Baldwin-Lomax turbulence model for projectile flow. Default is .TRUE.
BARO	Logical sentinel for pressure diffusion (if .TRUE.) with multiple species gas. Default value is .FALSE.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
CHOW	Logical flag to activate Chow wake mixing turbulence model for projectile flow. Default is .TRUE.
CLENG	Reference length in meters or feet. See IUNITS. No default value.
CONSRV '	Logical sentinel to indicate strong-conservation form of the governing P.D.E.'s (if .TRUE.) or weak-conservation form of the governing P.D.E's (if .FALSE.). Default value is .FALSE.
CONSRVF(IEQ)	Logical sentinel to indicate conservation form of the convective terms (if .TRUE.). Default value is NEQS * .TRUE.
CPINF	Reference specific heat at constant pressure. Calculated by the code.
CTBAF1	Turbulence model constant for freestream mixing length. Default value is 0.09.
CTBAF2, CTBAF3, CTBAF4	Mixing length turbulence model constants for boundary layer flows. Default values are 0.12, 0.81 and 2.0, respectively.
CTRB1	Constant C_1 in the turbulence dissipation rate equation. Default value is 1.43.
CTRB2	Constant C_2 in the turbulence dissipation rate equation. Default value is 1.92.
CTRBMU	Constant C_μ in Prandtl-Kolmogorov turbulent viscosity relation. Default value is 0.09.
CTWO	Coefficient of dilitation stress terms ($\nabla \cdot U$). Default value is 2/3.
CVIS1, CVIS2	Sutherland's law constants for molecular viscosity. Default values for air viscosity are 2.27×10^{-8} kg/(m − s − $(K)^{\frac{1}{2}}$ and 198.6 K.
CVK	Von Karman constant. Default value is 0.4.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
EFFDIFF	Logical sentinel to activate (if .TRUE.) the effective molecular diffusion coefficient model for a multiple species gas. Default value is .FALSE.
GAMMA	Ratio of specific heats (frozen). Default value is 1.4.
HTOT	Nondimensional total enthalpy for boundary conditions and for IHSTAG = 2 option. Calculated by the code using TSTAT1 and USTAT1.
IHSTAG	Enthalpy option. IHSTAG = 0: Energy equation is formulated in terms of a static enthalpy. IHSTAG = 1: Energy equation is formulated in terms of stagnation enthalpy. IHSTAG = 2: Constant stagnation enthalpy is assumed. Default value is 0.
INCOMP	Sentinel for incompressible equation of state. INCOMP = .TRUE.: Incompressible option. INCOMP = .FALSE.: Compressible option. Default value is .FALSE.
INJDAMP	Logical sentinel to activate (if .TRUE.) the turbulence damping model for injection driven flows. See subroutine SORCH1 and SORCH2 for cartesian and cylindrical-polar formulations, respectively. Default value is .FALSE.
ISETKE	First time step when k-ϵ partial differential equations are solved. Default value is 999999999.
ISETKEF	Last time step when the k-ϵ partial differential equations are solved with frozen fluid dynamic variables. When ISETKEF is set, ISETKE must be less than ISETKEF. Default value is −999999999.
ISMOOTH	Sentinel to activate turbulent viscosity algebraic smoothing option. Presently valid only for cascade version (see JGRID). ISMOOTH = 0: No smoothing. ISMOOTH = N: Smooth effective viscosity N times. Default value is 0.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
ISPREAD	Flow field initialization option. See Section 6. ISPREAD = 0: Set stagnation conditions. ISPREAD = 1: User supplied initialization in subroutine SPREAD. ISPREAD = 2: Turbulent pipe flow initialization in subroutine SPREADTP. ISPREAD = 3: External or cascade flow initialization in subroutine SPREAD2. Default value is 0.
ISWIRL	Swirl equation option (not needed for TWOD = .FALSE. or ICOORD = 0). ISWIRL = 0: No swirl equation is solved. ISWIRL = 1: Swirl equation is solved - used for two-dimensional cases with axisymmetric rotation. ICOORD must equal 1.
IUNITS	System of units utilized - see Table 2. IUNITS = 1: SI IUNITS = 2: English Default value is 1.
IVISC	Viscosity model option. IVISC = 1: Constant viscosity. Input of reference viscosity MUINF is required. IVISC = 2: Molecular viscosity determined by LAMVIS. IVISC = 3: Mixing length turbulent viscosity with molecular viscosity determined by LAMVIS. IVISC = 4: k–ℓm turbulent viscosity with molecular viscosity determined by LAMVIS. IVISC = 5: k-ϵ (Jones-Launder) turbulent viscosity with molecular viscosity determined by LAMVIS. IVISC = –5: Solve k-ϵ equations only with frozen fluid dynamics. Default value is 3.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
KAPPA	Molecular thermal conductivity option. KAPPA = 1: Molecular thermal conductivity calculated from Prandtl number, $\kappa_\ell = \mu_\ell C_p/Pr_\ell$, where Pr_ℓ = PRLAM is an input variable. KAPPA = 2: Molecular thermal conductivity specified by user in AC(TCO, IADD). No turbulent conductivity is included. KAPPA = 3: Molecular thermal conductivity calculated using Euken's formula for a multiple species gas. Also, requires LAMVIS = 3 option. Note: turbulent conductivity is determined from $\kappa_T = \mu_T C_p/P_{r_T}$, where Pr_T = PRTURB is an input variable. Default value is 1.
LAMVIS	Molecular viscosity option. LAMVIS = 1: Constant molecular viscosity equal to VISCIN. Input of reference viscosity MUINF is required. LAMVIS = 2: Sutherland's law for the molecular viscosity. Constants CVIS1 and CVIS2 are required. LAMVIS = 3: Wilke's formula for the molecular viscosity of a multiple species gas. Default value is 2.
LEWIS	Lewis number option for multiple species energy equation. LEWIS = 0: Normal calculation of interdiffusional energy flux assuming equal binary diffusion coefficients, and equal molecular and turbulent Schmidt numbers, SCLAM and SCTURB. See also EFFDIFF. LEWIS = 1: Assumes unity Lewis number (Le = Pr/Sc) in calculating the interdiffusional energy flux. This is less expensive in CPU time than LEWIS = 0. Default value is 0.
MINF	Free stream Mach number. Default value is 0.0.
MUINF	Dimensional free stream viscosity. Input is required for LAMVIS = 1. See Table 2. Default value is 0.0.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
NXINF, NYINF, NZINF	Computational grid points for free stream location. Used in calculation of free stream pressure and dynamic head for pressure coefficient output in cascade flow (JGRID = 1, 2, 3 or 4).
PINF	Dimensional free stream static pressure. See Table 2. Default value is 0.0.
PNOM	Nominal pressure for perturbation pressure calculation. Default value is 0.0.
PRLAM	Molecular Prandtl number. Default value is 0.73.
PRTURB	Turbulent Prandtl number. Default value is 1.0
PSTAT1	Nondimensional free stream static pressure used for calculating the boundary condition value of total pressure (PTOT). Default value is 1.0.
PTOT	Nondimensional total pressure for boundary conditions. Calculated by the code using PSTAT1, TSTAT1 and USTAT1.
PZERO	Dimensional free stream stagnation pressure. See Table 2. Default value is 0.0.
REPL	Dimensional Reynolds number per unit length. Default value is 0.0.
REY	Reference Reynold number. Calculated by the code.
RHOINF	Dimensional free stream density. Calculated by the code. Default value is 0.0.
RHO1	Not used.
SCLAM	Molecular Schmidt number. Default value is 0.73.
SCTURB	Turbulent Schmidt number. Default value is 1.0.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
SIGE	Turbulence energy dissipation rate equation constant for diffusion (like the Prandtl number for thermal diffusion). Default value is 1.3.
SIGK	Turbulence kinetic energy equation constant for diffusion (like the Prandtl number for thermal diffusion). Default value is 1.0.
SOUND	Reference value of sound speed. Calculated by the code.
TEDMIN	Minimum value of turbulence energy dissipation rate (ϵ) allowed by the code. Default value is 1.0×10^{-12}.
TINF	Dimensional free stream static temperature. See Table 2. Default value is 0.0.
TKEMIN	Minimum value of turbulence kinetic energy (k) allowed by the code. Default value is 1.0×10^{-12}.
TSTAT1	Nondimensional free stream static temperature used for calculating the boundary condition values of HTOT, TTOT and PTOT. Default value is 1.0.
TTOT	Nondimensional total temperature for boundary conditions. Calculated by the code using TSTAT1 and USTAT1.
TWOPHS	Logical sentinel to activate (if .TRUE.) the Lagrangian two-phase flow option. See Namelist $TWOP.
TZERO	Dimensional free stream total temperature. Default value is 0.0.
UINF	Dimensional free stream velocity. Default value is 0.0.
USTAT1	Nondimensional free stream velocity used for calculating the boundary condition values of HTOT, TTOT and PTOT.

NAMELIST $READ3 (continued)

VARIABLE NAME	DESCRIPTION
WMINBC	Minimum absolute value of inflow velocity allowed for certain boundary conditions (IEQBC = 19, 20, 21). A small positive value is required when starting from stagnation conditions (e.g., 1.0×10^{-5}). Default value is 0.0.
WTREF	Reference mixture molecular weight. Default value is 28.966.

NAMELIST $READ4 — Time Step and Print Control

DELDT	Factor for increasing or decreasing the time step DT when IDTADJ = 1. Default value is 1.25.
DELDTAU	Factor for increasing or decreasing the pseudo time step DTAU when KDTADJ = 1. Used only when iteration within each physical time step is selected (KITER > 1) for transient problems. Default value is 1.2.
DIMPPT	Logical sentinel to activate (if .TRUE.) dimensional output of all dependent and derived variables. See IUNITS and IVARPR, Tables 2 and 5. Default value is .FALSE.
DT	Initial nondimensional time step. If DT is omitted on a restart, DT will be set to the value at termination of last run. No default value.
DTAU	Nondimensional pseudo time step for transient iteration within each physical time step. See KITER. Default value is 1.0.
DTAUMAX	Maximum permitted value of DTAU. Default value is 3.0.
DTAUMIN	Minimum permitted value of DTAU. Default value is 1.0
DTMAX	Maximum nondimensional time step for this run. No default value.
DTMIN	Minimum nondimensional time step for this run. No default value.
DTM1	Value of DT at previous time step (IDT − 1) for transient problems. Set by the code.

NAMELIST SREAD4 (continued)

VARIABLE NAME	DESCRIPTION
EPSS	Convergence tolerance for Newton-Raphson iterations throughout the code. Default value is 1.0×10^{-8}.
IDT	Time step number. Set by the code.
IDTADJ	Time step control. IDTADJ = 0: Constant DT is used for this run. IDTADJ = 1: Time step adjusted. If maximum relative change per time step in any flow variable is less than PCNT1, DT is multiplied by DELDT to a maximum of DTMAX. If maximum relative change in any flow variable is greater than PCNT2, DT is divided by DELDT**2 to a minimum of DTMIN. IDTADJ = 2: Time step is cycled between DTMIN and DTMAX over NTSTEP time steps. Default value is 1.
IDTGCP	Time step interval for moving geometry printout. Default value is 1.
IDUMP1	Sentinel for initial printout of geometry and fluids variables. IDUMP1 = 0: no printout. IDUMP1 = 1: printout provided. Default value is 0.
IGPRT(IV)	Print control flag for variable IV - see Table 4. Default values are NGVMAX*0.
IPLOT	Time step interval when plot file will be written. If IPLOT = 999, plot file will be written on first initial - or restart step and calculation terminated. Default value is 0.0.
IPLOTF	Number of plot files written beginning with the initial run. Set by the code.
IPRINT	Time step interval for intermediate fluid variable printout. IPRINT = 0: Print only at last time step. Default value is 0.
IPRINTA	Time step interval for spatial averaged quantity printout. Default value is 0 (print only at last time step.)
IRESTF	Number of restart files written beginning with the initial run. Set by the code.

NAMELIST $READ4 (continued)

VARIABLE NAME	DESCRIPTION
ISSCNT	Time step interval for the time step summary print. Must be greater than 0. Default value is 1.
ITCALC	Time step selection sentinel. ITCALC = 0: No spatial time step variation. 1: No spatial time step variation with Mach number conditioning. 2: Spatial time step variation (geometric). 3: Spatial time step variation (geometric) with Mach number conditioning. 4: Spatial time step variation. 5: Spatial time step variation with Mach number conditioning.
ITEST	Time step interval for steady state convergence test. ITEST = 0 disables test. Default value is 1.
ITPLT	Counter for time history data storage. Set by the code.
IVARPL(IP)	Plot file variable. See Table 5. IVARPL(IP) = −1: Omit variable IV from plot file. IVARPL(IP) = 0: IP^{th} variable from Table 5 is written on plot file. IVARPL(IP) = IV: Variable number IV(Table 1) is the IP^{th} variable written on plot file. See IVARSP for multiple species problems. Default values are (NCOEQM + 10), which causes the variables in Table 5 to be written. See IVARSP for multiple species problems.
IVARPR(IV)	Print control flag for fluid dynamic variable IV - see Table 6. Print of variable IV is disabled if IVARPR(IV) = 0. Default values are 60 * 0.
IVARPRA(IV)	Print control flag for spatially averaged fluid dynamic variable IV - same variable correspondence as for IVARPR - Table 6. Default values are 60 * 0.
IVARSP(ISP)	Species number ISP plot file control. IVARSP(ISP) = 0: Do not write species ISP on plot file. IVARSP(ISP) = 1: Write species ISP on plot file. Default values are NCOEQM * 0.

NAMELIST $READ4 (continued)

VARIABLE NAME	DESCRIPTION
JPR	Output control flag - see IVARPR. JPR = 1: Print ($y^2 - y^3$) planes at constant y^1. JPR = 2: Print ($y^1 - y^3$) planes at constant y^2. JPR = 3: Print ($y^1 - y^2$) planes at constant y^3. Used only for 3-D problems, TWOD = .FALSE. For 2-D problems $y^1 - y^3$ planes are always printed. Default value is 3.
JPRA	Output control flag - see IVARPRA. Same meaning as JPR. Default value is 3.
KDTADJ	Pseudo time step control for transient problems. See KITER. KDTADJ = 0: Constant dTAU. KDTADJ = 1: DTAU is adjusted. If maximum relative change per iteration in any flow variable is less than PCNTD1, then DTAU is multiplied by DELDTAU to a maximum of DTAUMAX. If the minimum relative change is greater than PCNTD2, then DTAU is divided by DELDTAU ** 2 to a minimum of DTAUMIN. KDTADJ = 2: Pseudo time step DTAU is cycled between DTAUMIN and DTAUMAX over KTSTEP iterations. Default value is 1.
KITER	Number of inner iterations for each physical time step for transient problems. Default value is 1.
KITPRT	Time step interval for inner iteration summary print. Used only if KITER > 1. Default value is 100.
KTAPE	Plot file unit number. Default value is 1.

NAMELIST $READ4 (continued)

VARIABLE NAME	DESCRIPTION
KTCALC	Pseudo time step selection sentinel for transient problems. KTCALC = 0: Nonspatially varying pseudo time step, DTAU. KTCALC = 2: Spatially varying pseudo time step - geometric option. KTCALC = 4: Spatially varying pseudo time step - eigenvalue option. Default value is 4.
KYCYST	Time step number at which pseudo time step cycling option (KDTADJ = 2) will be initiated. Default value is 999999.
KTSTEP	Number of pseudo time steps for cycling option, KDTADJ = 2.
LYDOUB(LY)	3-D double-valued point print control flag. LYDOUB(LY) = 0 disables all double-valued point print for the plane y^2 = LY. Default values are NY * 1.
MASMOLO	Logical multiple species output sentinel. MASMOLO = .FALSE.: Mole fraction output. MASMOLO = .TRUE.: Mass fraction output.
MEQS2	Total number of equations being solved. Set by the code.
NFGRP(IEQ)	Fluid dynamic equation "group" flag to control equation coupling at each time step. NFGRP(IEQ) = IG indicates that equation IEQ is in group IG, and will be coupled to all other equations in group IG. See Table 1. Default values are 5 * 1, 2 * 2.
NFPLOT	Total number of plot files written. Set by the code.
NITER	Maximum number of Newton-Raphson iterations allowed. Used for multiple species gases (NCOEQ > 0) or non-calorically perfect gases (CPVAR = .TRUE.). Default value is 50.
NLWR(NL)	Namelist output control sentinel. See Table 7 for namelist numbering. NLWR(NL) = 0: Do not print namelist number NL. NLWR(NL) = 1: Print namelist number NL. Default vlaues are 5 * 1, 7 * 0.

NAMELIST $READ4 (continued)

VARIABLE NAME	DESCRIPTION
NPADI	Number of groups of coupled equations. Set by the code. See NFGRP and NCOGRP ($CHEM).
NT	Number of time steps to be run. Default value is 0.
NTCYST	Time step number to begin time step cycling option, IDTADJ = 2. Default value is 999999.
NTREST	Time step interval for writing a restart file. Restart file is always written at last step. Default value is 999999999.
NTSTEP	Number of time steps per cycle for IDTADJ = 2 option. Default value is 4.
PCNTD1	Lower percentage change per iteration in dependent variables for pseudo-time step control option, KDTADJ = 1. Default value is 0.04.
PCNT1	Lower percentage change per time step in dependent variables for time step control option IDTADJ = 1. Default value is 0.04.
PCNTD2	Upper percentage change per iteration in dependent variables for pseudo time step control option, KDTADJ = 1. Default value is 0.06.
PCNT2	Upper percentage change per time step in dependent variables for time step control option, IDTADJ = 1. Default value is 0.06.
SSEPS	Criterion for maximum change in dependent variable for steady state convergence test. Default value is 1.0×10^{-5}.
SSEPSD	Iteration convergence criterion for maximum change in dependent variable. Used only when KITER > 1. Default value is 1.0×10^{-5}.
TEXIT	CPU time (in seconds) allowed for termination of a run which has reached the CPU time limit specified on the JOB card. Not used under Unicos operation system at present. Default value is 0.

NAMELIST $READ4 (continued)

VARIABLE NAME	DESCRIPTION
THREELEV	Logical sentinel to activate (if .TRUE.) the three time level transient iteration scheme. See KITER. This option requires an additional time level of storage for all dependent variables in the AC array. See parameter NVLAC. Default value is .FALSE.; however, this option is recommended for transient accuracy improvement.
TIMSCL(IEQ)	Time scaling factor for equation IEQ. See Table 1. This factor scales the time step for equation IEQ, and may be used to improve convergence for certain problems. Typically used for the energy and turbulence (k-ϵ) equations. Default values are NEQS * 1.0.
TTIME	Nondimensional time from the initial run. Set by the code. Not meaningful unless ITCALC = 0.

NAMELIST $READ5 — Boundary Condition Input - See Table 8

BLPCNT	Boundary layer thickness used for flow field initialization in subroutine SPREAD2. Default value is 0.01.
DTROFF(KSURF,IDIR)	Control parameter for porous wall normal injection for direction IDIR. See Figure 4. Default values are all 0.0.
DTRON(KSURF,IDIR)	Control parameters for porous wall normal injection for direction IDIR. See Figure 4. Default values are all 0.0.
IBOUND(KSURF,IDIR)	Type of boundary on surface number KSURF corresponding to direction IDIR. See Figure 2. IBOUND(KSURF,IDIR) = 0: Periodic boundary 1: Inlet 2: Symmetry line or plane 3: Exit 4: Wall Default values are all −1.
IEQBC(KSURF,IDIR,IEQ)	Boundary condition on surface KSURF corresponding to direction IDIR associated with equation IEQ. Default values are set in subroutine BCSET depending on input values of IBOUND(KSURF,IDIR). See Table 8.

NAMELIST $READ5 (continued)

VARIABLE NAME	DESCRIPTION
INJMF1(KSURF,IDIR,ICO)	Injection mass or mole fraction of species ICO for direction IDIR on surface KSURF at time step number INJT1. Used for boundary condition IEQBC = 34 (mass fraction) or 35 (mole fraction) to vary the species distribution on surface KSURF from INJMF1 at time step INJT1 to INJMF2 at time step INJT2. No default values.
INJMF2(KSURF,IDIR,ICO)	Injection mass or mole fraction of species ICO for direction IDIR on surface KSURF at time step number INJT2. See INJMF1. No default values.
INJT1	Initial time step for species injection boundary conditions, IEQBC = 34 and 35. Species mass/mole fraction is set equal to INJMF1 for IDT ≤ INJT1. Default value is 0.
INJT2	Final time step for species injection boundary conditions, IEQBC = 34 and 35. Species mass/mole fraction is set equal to INJMF2 for IDT ≥ INJT2. Default value is 0.
IPERBC(IADI)	Periodic boundary sentinel. IPERBC(IADI) = 0: Normal boundaries for direction IADI. IPERBC(IADI) = 1: Periodic boundaries for direction IADI. IEQBC not required for periodic boundaries. Not valid for IADI = 3. Default values are 3 * 0.
LGEDGE(IG,IE,KSURF)	Boundary layer edge (grid point) location for grid point number IG on surface KSURF. The index IE = 1 or 2 indicates the lower and upper boundary layer edge, respectively. Used for boundary conditions IEQBC = 19, 20 and 21. User modification of subroutines PRESSET and BC is required in 3-D. Default values are all 0.
LYSYM	Sentinel for 3-D symmetry plane in cylindrical-polar coordinate system. LYSYM is the θ-direction (y^2) coordinate index of the grid line normal to a plane of symmetry in ($r - \theta$) plane. Default value is 0.
NPRES1	Time step when pressure has a value of PRES1 - IEQBC = 5. Default value is 1.

NAMELIST $READ5 (continued)

VARIABLE NAME	DESCRIPTION
NPRES2	Time step when pressure has a value of PRES2 - IEQBC = 5. Default value is 10000.
NTWALL1	Time step when temperature has a value of TWALL1 - IEQBC = 11. Default value is 1.
NTWALL2	Time step when temperature has a value of TWALL2 - IEQBC = 11. Default value is 10000.
NVROT1	Time step when rotational velocity is VROT1 - IEQBC = 32. Default value is 1.
NVROT2	Time step when rotational velocity is VROT2 - IEQBC = 32. Default value is 10000.
PAMB(KSURF,IDIR)	Nondimensional pressure on surface number KSURF associated with direction IDIR - IEQBC = 4.
PERM(KSURF,IADI)	Permeability of surface KSURF in direction IADI for IEQBC = 40. PERM $\geq$ 0.0: ρu_{normal} = PERM * (P − PPL) PERM < 0.0: ρu_{normal} = PPL No default values.
PPL(KSURF,IADI)	Plenum pressure mass flux for surface KSURF in direction IADI − IEQBC = 40. See PERM. Note PPL is always input as a positive nondimensional number.
PRES1	Corresponding pressure at time step NPRES1. Default value is 1.0.
PRES2	Corresponding pressure at time step NPRES2. Default value is 1.0.
RADPRF(IG1,IG2)	Nondimensional radial velocity profile with wall injection - IEQBC = 42. IG1 and IG2 are the grid points on the boundary surface. Input of UWALL is required. No default values.
SCLPRO(JG,I)	Normalized velocity profiles, I = 1 − 3 for flow field initialization (subroutine SPREAD2).

NAMELIST $READ5 (continued)

VARIABLE NAME	DESCRIPTION
TROFF(KSURF,AIDI)	See DTROFF.
TRON(KSURF,IADI)	See DTRON.
TWALL(KSURF,IDIR)	Nondimensional temperature on surface number KSURF associated with direction IDIR. Default values are 0.0.
TWALL1	Corresponding nondimensional temperature at time step NTWALL1. Default value is 1.0.
TWALL2	Corresponding nondimensional temperature at time step NTWALL2. Default value is 1.0.
USURF(I)	Moving surface velocity components (I = 1, 2 and 3) – IEQBC = 70. Default values are 3 * 0.0.
UWALL	Wall velocity for radial velocity profile with wall injection - IEQBC = 42. See RADPRF. No default value.
VROT1	Corresponding nondimensional rotational velocity at time step NVROT1. Default value is 0.0.
VROT2	Corresponding nondimensional rotational velocity at time step NVROT2. Default value is 0.0.

NAMELIST $READ6	Pressure Input
BLPROF(IG1,IG2)	Normalized boundary layer velocity profile for use with the "2-layer" boundary condition, IEQBC = 19. IG1 and IG2 are the grid points in the boundary planes. See Section 6 and LGEDGE. No default values.
PRESS1(IG1,IG2,IBC)	Nondimensional static or total pressure input on surface IBC (in general can be a function of grid points IG1 and IG2). Used in conjunction with boundary conditions IEQBC = 17 or 18. In some cases, it may be desirable to set up PRESS1 values in subroutines SPREAD and PRESSET.

NAMELIST $MISC — Miscellaneous Input

Problem Dependent Input Setup at User Discretion. (See subroutine RDLIST).

NAMELIST $CHEM — Multiple Species and Chemistry Input

VARIABLE NAME	DESCRIPTION
ANU1(IS,IR)	Left-hand side stoichiometric coefficients ($\nu'_{s,r}$) of species IS in reaction number IR. See Appendix A, Eq. (A39). Default values are all 0.0.
ANU2(IS,IR)	Right-hand side stoichiometric coefficient ($\nu''_{s,r}$) of species IS in reaction number IR. See Appendix A, Eq. (A39). Default values are all 0.0.
CHEMSCL(IEQ)	Time step scaling parameter for species conservation equation IEQ. Used when chemical source terms are present. Default values are NCOEQM * 0.0.
CONREF	
CPVAR	Logical sentinel which indicates temperature dependent specific heat, if .TRUE. Also activated by JANAFOPT > 0. Default value is .FALSE.
CRRATE(IR)	Dimensional chemical reaction rate coefficient for reaction number IR. Units are $(m^3/molecule)^n/sec$, where n is the order of the reaction. Reaction rate is of the form: CRRATE * T^{RPOWER} exp[−TACTIV/T]. Default values are NCOEQM * 0.0.
DREF	
EACTIV(IR)	Nondimensional activation energy. Calculated by code.
IBRES(IR)	Backward reaction sentinel for reaction IR. IBRES(IR) = 0: No backward reaction allowed. IBRES(IR) = 1: Backward reaction allowed. Default values are all 0.
IEQLCHEM	Not used at present.
IREACM(IR)	M-body sentinel for reaction IR. IREACM(IR) = 1: Indicates the presence of a passive reactant (M) in reaction IR. Default values are all 0.

NAMELIST $CHEM (continued)

VARIABLE NAME	DESCRIPTION
JANAFOPT	Sentinel for NASA LeRC JANNAF data base option. JANAFOPT = 0: Data base not used. JANAFOPT = 1: Standard data base file generated by CET86 code (Ref. C-2) is used. JANAFOPT = 2: Special data file generated by modified CET86 code is used. Activates equilibrium chemistry option. JANAFOPT = 3: Data base file not used. Thermochemical data is read only from Namelist $JANAFNL (Section 10).
LISTIN	Logical sentinel to activate, if .TRUE., the list directed input for species data. Needed only on a restart run to change species list. The list directed input, Section 10, is mandatory if CPVAR = .TRUE. or JANAFOPT = 1 or 3. Default value is .FALSE.
NCOEQ	Number of species (component) equations to be solved. Default value is 0.
NCOGRP(IS)	"Group" number for species (component) number IS. All equations with the same group number (NFGRP or NCOGRP) are solved as a coupled system at each time step. NFGRP is described in NAMELIST $READ4. Also NCOGRP is input in the list directed input which takes precedence over this namelist value. See Section 10. Default values are NCOEQM * 3.
NTRNG	Number of temperature ranges in the JANNAF data base curve fit coefficients used in CMINT. Set by the code. See NTRANGE in Namelist $JANAFNL (Section 10).
PRPOSN	
ROCKET	Logical sentinel to activate rocket motor internal flow option which sets the other required option flags required by CMINT. The user must also set NCOEQ = 1 to solve a metal species conservation equation for a metallized propellant. Default value is .FALSE.
RPOWER(IR)	Nondimensional temperature exponent in chemical reaction rate for reaction IR. See CRRATE and TACTIV. Default values are NCOEQM * 0.0.
TACTIV(IR)	Dimensional activation temperature in chemical reaction rate for reaction IR. See CRRATE and RPOWER. Default values are NCOEQM * 0.0.

NAMELIST $TWOP — Lagrangian Two-phase Flow Input

VARIABLE NAME	DESCRIPTION
CFRICT	Friction coefficient for wall reflections.
CPAR	Specific heat for the particle.
CPRESF	Coefficient for effective pressure of particle motion. CPRESF = 0: Particle motion not affected by pressure gradient. −1: Affected by full pressure gradient.
CREST	Restitution coefficient for wall reflection.
DDYX22	dY2/dX2 where dY2 is the second computational coordinate, and dX2 is the second physical coordinate. Used to simulate 3-D particle trajectories in a 2-D or in an axisymmetric geomtry. 2-D: DDYX22 = number of mesh cells per unit length in the third direction. Axisymmetric: DDYX22 = number of mesh cells per 2π radians in the circumferential direction.
ENRINT	AC array index for energy interaction source term (integer).
FROUDI	Inverse Froude number (based on acceleration due to gravity).
GAXIS(I)	Array of direction cosines for the (gravitational) acceleration vector, I = 1, 2, 3.
ICLVLS	AC array index for cell volume.
IDEFBD	Flag to signify boundary definitions. DEFBND subroutine uses this flag to install appropriate boundary definitions.
IDTAV1	Time step number to start source term averaging.
IDTAV2	Time step number to end source term averaging.
IDTSAM	Time step number to start sampling of discrete phase properties. Store sample particle properties in AC array.
IOPTR	Flag to specify trace information writing (see subroutine WTRACE).
IPNDBG	Debug particle number: dump all information for particle number IPNDBG.

NAMELIST $TWOP (continued)

VARIABLE NAME	DESCRIPTION
IPNO	Absolute particle number, i.e. number of particles injected so far including those that have already evaporated/left domain. Used to track a single particle. (Don't set this unless you know what you are doing).
ISAMPL	Array of AC array indices corresponding to discrete phase ("sampled") properties.
ISEED	Seed for random number generation. Don't specify this unless you know what you are doing. Used to store seed when ending a calculation that will be restarted later.
ITRACE	An array of particle numbers that will be traced in subroutine PTRACE.
JSAMPL	Array indicating which properties are to be sampled.
MASINT	AC array index for mass interaction source term (to be added to Eulerian mass source term).
MFINT(•)	AC array indices for mass fraction interaction source term(s). The number of slots depends on the number of species being solved.
MOMIN1	AC array index for x momentum interaction source term.
MOMIN2	AC array index for y momentum interaction source term.
MOMIN3	AC array index for z momentum interaction source term.
MOTION	Flag to specify type of particle motion. MOTION = 0: 3-D geometry and 3-D particle motion = 1: 2-D geometry and 2-D particle motion = 2: 2-D geometry and 3-D particle motion = 3: axisymmetric geometry and 3-D particle motion.
NPAR	Current number of particles in the PAR array.
NPULSE	Number of injection pulses per Lagrangian time step.
NSAMPL	Number of discrete phase properties to be sampled.
NTRACE	Number of trajectories to be stored in subroutine PTRACE.

NAMELIST $TWOP (continued)

VARIABLE NAME	DESCRIPTION
NUNTR1	Logical unit number to write trajectory traces. Default value is 26.
PCRNEX	Flag to determine the treatment of outward facing corners (convex) for reflections. PCRNEX = 0: More exact (and expensive) treatment. = −1: Approximate but preferred treatment. Default value is 0.
PCRNIN	Flag to determine the treatment of inward facing corners (concave) for reflections. PCRNIN = 0: More exact (and expensive) treatment. = −1: Approximate but preferred treatment. Default value is 0.
PDTFAC	Lagrangian time step Δt; DTPART is set equal to PDTFAC.
PLTHT	Latent heat of vaporization for the particle.
PRFAC	Fraction of the particle radius used to determine when the particle reflects from a wall. Default value is 0.
PSATM	Maximum possible saturation pressure for liquid (hypothetical saturation pressure at ∞ Kelvin).
PSTEP	Inverse of the fraction of computational secondary cell traversed in one sub time step. Default value is 1.0.
PSWTCH	Particle size switch (set to zero). PSWTCH = 0: Set DXP = 0, compute DYPBOX(K) a priori using the transformation properties at the particle location. = 1: Set DYPBOX(K) = 0, use DXP to compute DYK locally, for each solid wall Y(K) = constant.
RATINJ	Mass injection rate (set in subroutine INJECT).
RDROP	Particle radius (set in subroutine INJECT).
RHOP	Particle density.
TEVRAT	Total evaporation rate for all particles present in field at a given time (calculated).

NAMELIST $TWOP (continued)

VARIABLE NAME	DESCRIPTION
TINJCT(•)	Particle temperature at injection (set in subroutine INJECT).
TPMASS	Total particulate phase mass in domain.
UINJCT(•)	Array of particle u velocities at injection (set in subroutine INJECT).
VINJCT(•)	Array of particle v velocities at injection (set in subroutine INJECT).
WINJCT(•)	Array of particle w velocities at injection (set in subroutine INJECT).
VF	AC array index for void fraction which is passed to Eulerian code (integer).
Y1INJ(•)	Array of y^1 coordinate locations for injection (set in subroutine INJECT).
Y2INJ(•)	Array of y^2 coordinate locations for injection (set in subroutine INJECT).
Y3INJ(•)	Array of y^3 coordinate locations for injection (set in subroutine INJECT).
Y2PRD	The 'reduced' domain corresponds to the region $-Y2PRD < y^2 - 1 < Y2PRD$ in the computational domain. therefore, the region N is the region $(2*N - 1)*Y2PRD < y^2 - (2*N + 1)*Y2PRD$ in the computational domain.

NAMELIST $M864 — Projectile Flow Namelist (follows $TWOP)

VARIABLE NAME	DESCRIPTION
IBASE	Flag to indicate projectile base and nose configuration. See Figure 5 for schematic of physical and computational domain. Default is 0 (implies extended string configuration).

NAMELIST $M864 (continued)

VARIABLE NAME	DESCRIPTION
IBASEOPT	Flag for projectile base flow field initialization option. See subroutine BASESTRT, which writes and reads a file (unit 23) containing upstream profile data for the base region calculation initialization. LZMATCH is also required. Default value is 0 (IBASEOPT is set by the code).
IDTPLT	Time step interval for writing base drag and convergence history information to unit 24. Default value is 99999999.
LXLEAD, LZLEAD	Computational grid point number for "leading edge" (projectile nose corner or upstream boundary). Set by the code.
LXTRAIL, LZTRAIL	Computational grid point number for "trailing edge" (projectile base corners). Set by the code.
LZMATCH	Axial direction grid point number at which upstream profile data is written to unit 23. Default value is 0. Set this on a forebody run.
MCASE	Defines projectile configuration for subroutine CPPLOT. Set by the code. See subroutine ARCLEN.
NXZDL	Not used.
PROJTILE	Logical flag to activate projectile flow field option. Default value is .TRUE.
XLEAD	Physical radial location of "leading edge" (projectile nose corner on upstream boundary). Set by the code.
ZLEAD	Physical axial location of "leading edge" (projectile nose corner on upstream boundary). Set by the code.
XTRAIL	Physical radial location of "trailing edge" (projectile base corner). Set by the code.
ZTRAIL	Physical axial location of "trailing edge" (projectile base corner). Set by the code.

Intentionally left blank.

10.0 Multiple Species Gas Input

The following additional input is needed for a multiple species problem: the required variables are read via a list directed read (free format) with seven items per record. An 'END' card is required to terminate the list directed input.

List Directed Input

	Item	Units	Example	Data Type
1.	Species Name	None	'H2O'	Character
2.	Species Molecular Weight[1]	kg/kg-mole	0.0	Real
3.	Collision Diameter of Molecule (σ)[2]	Angstroms	0.0	Real
4.	Characteristic Interaction Energy (ϵ/K)[2]	Kelvin	0.0	Real
5.	Reference Mass/Mole Fraction[3]	None	±1.0	Real
6.	Species Equation Group (NCOGRP)[4]	None	3	Integer
7.	Data Base Source[5]	None	'JANNAF'	Character

The 'END' card format is:

'END' 0. 0. 0. 0. 0 'X'

Note that character data must be enclosed in single quotes. The species names must correspond to those on the NASA LeRC JANNAF data base file (Ref. C-2).

---

[1] The species molecular weight value will be obtained from the JANNAF data base file. The value entered here is only for species which are not on the data base. Do not leave this field blank.

[2] Used for the species molecular (laminar) viscosity model. See Ref. C-4, p. 22 ff.

[3] A positive value indicates mass fraction; a negative value indicates mole fraction.

[4] See NAMELIST $CHEM for description.

[5] If a species is not on the data base, this field must be changed to 'NONE' and appropriate data must be input in the NAMELIST $JANAFNL, described below.

The reference mass fraction is used to calculate the reference molecular weight for the problem under consideration. These values must not be changed on a restart run.

NAMELIST $JANAFNL — Optional Thermochemical Data

VARIABLE NAME	DESCRIPTION
AISPEC(I,IT,IS)	JANNAF curve fit coefficients, $a_{I,IS}$, Eqs. (C2, C4, C6), for species number IS and I = 1 – 7. The temperature range IT = (1, 2, ..., NTRANGE) conforms to the JANNAF data base standard; i.e., IT = 1 is the high range, and IT = 2 is the low range. CMINT optionally adds a low range and a high range by assuming constant specific heat and matching the data base. See TDFITD, TDMAXD and TDMIND.
NTRANGE	Number of temperature ranges for which thermochemical data coefficients are supplied, either from the JANNAF data base or via input. See AISPEC. Default value is 2.
SMW(IS)	Molecular weight of species number IS.
TDFITD(ITB)	Dimensional temperature bounds (Kelvin) for thermochemical data coefficients, AISPEC(I,IT,IS). For temperature range IT, the bounding temperatures are TDFITD(IT) and TDFITD(IT+1). The default values are 5000.0, 1000.0, and 300.0 Kelvin.
TDMAXD	Maximum dimensional temperature (Kelvin) for the data base extension at constant specific heat. The high temperature range will extend from TDFITD(1) to TDMAXD. If TDMAXD ≤ TDFITD(1), then the high temperature range extension is not added. Default value is 0.0 Kelvin.
TDMIND	Minimum dimensional temperature (Kelvin) for the data base extension at constant specific heat. The low temperature range will extend from TDMIND to TDFITD(NTRANGE+1). If TDMIND ≥ TDFITD(NTRANGE+1), then the low temperature range extension is not added. Default value is 0.0 Kelvin.

NAMELIST $PARTBRL — Two-Phase Flow Input for AP Particle Injection (Subroutine INJECT) (Follows $JANAFNL)

VARIABLE NAME	DESCRIPTION
PARTMSF	Particle injection mass flux (fraction of injection gas mass flux). Default value is 0.01.
INCLZINJ	Axial direction grid point increment for particle injection. Particles will be injected at the base plus INCLZINJ. This is used to avoid injection in a fine grid region. Default value is 0.
NTYPE	Number of different particle types to be injected. Default value is 1 (Ammonium perchorate is assumed).
DMEAN	Mean diameter of particles, meters. Default value is 5.0 E-6.
RHOPD	Particle density, kg/m^3. Default value is 1950.
SIG	Standard deviation for diameter of injected particles, meters. Default value is 0.0.
FRAC(I)	Fraction of particles of each type. See NTYPE. Default value of FRAC(I) is 1.0.
Y1INMN	Computational coordinate minimum in radial direction for particle injection. Default value is 1.[6]
Y1INMX	Computational coordinate maximum in radial direction for particle injection. Default value is LBRKZ(I) - 1.
FINJN	Gas injection parameter (I). Set by the code in subroutine CPPLOT.
AREAB	Nondimensional projectile base area on injection surface. Set by the code in subroutine CPPLOT.

[6] Note that without turbulence the particle injection should start at Y1INMN = 2 to avoid particle accumulation at the stagnation point.

REFERENCES

1. Briley, W.R. and McDonald, H: On the Structure and Use of Linearized Block Implicit and Related Schemes. SRA Report R78-3, 1978, Journal of Comp. Physics, Vol. 34, No. 1, pp. 54-72, January 1980.

2. Briley, W.R. and McDonald, H.: Solution of the Multidimensional Compressible Navier-Stokes Equations by a Generalized Implicit Method. Journal of Comp. Physics, Vol. 24, pp. 372-297, August 1977.

3. Briley, W.R., McDonald, H. and Shamroth, S.J.: A Low Mach Number Euler Formulation and Application to Time Iterative LBI Schemes. AIAA Journal, Vol. 21, No. 10, pp. 1467-1469, October 1983.

4. Shamroth, S.J., McDonald, H. and Briley, W.R.: Prediction of Cascade Flow Fields Using the Averaged Navier-Stokes Equations. ASME Journal of Eng. for Gas Turb. and Power, Vol. 106, No. 2, pp. 383-390, April 1984.

5. Briley, W.R., Buggeln, R.C. and McDonald, H.: Solution of the Three-Dimensional Navier-Stokes Equations for a Steady Laminar Horseshoe Vortex Flow. SRA Report R84-920014-F, 1984.

6. Oh, Y.N.: An Analytic Transformation Technique for Generating Uniformly Spaced Computational Mesh. Final Report, NASA-Langley Research Grant NSG1087, October 1978.

7. Schlicting, H.: Boundary-Layer Theory. Sixth Edition, McGraw-Hill Book Company, pp. 192-193, 1968.

8. Maise, G. and McDonald, H.: Mixing Length and Kinematic Eddy Viscosity in a Compressible Boundary Layer. AIAA Journal, Vol. 6, No. 1, pp. 73-80, 1968.

Table 1. External Equation Numbers and Dependent Variable Numbers

Equation No. (IEQ)	Equation
1	x- or r-momentum
2	y- or θ-momentum
3	z-momentum
4	Continuity
5	Energy
6	Turbulence kinetic energy
7	Turbulence energy dissipation rate
8 - NEQS	Species mass fraction

Dependent Variable No.	Variable
1	x- or r-velocity component (u)
2	y- or θ-velocity component (v)
3	z-velocity component (w)
4	Density (ρ)
5	Enthalpy (h or h_0)
6	Turbulence kinetic energy (κ)
7	Turbulence energy dissipation rate (ϵ)
8 - NEQS	Species mass fraction (ω_i)

Table 2 - Reference Quantities - Units

Reference Quantity	FORTRAN Variable	Units	SI (IUNITS = 1)	English (IUNITS = 2)
ℓ_∞	CLENG	ℓ	m	ft
ρ_∞	RHOINF	$\frac{\text{mass}}{\ell^3}$	$\frac{\text{kg}}{\text{m}^3}$	$\frac{\text{slugs}}{\text{ft}^3}$
T_∞	TINF	Deg	K	°R
P_∞	PINF	$\frac{F}{\ell^2}$	$\frac{N}{m^2}$ (Pa)	$\frac{\text{lbf}}{\text{ft}^2}$
U_∞	UINF	$\frac{\ell}{t}$	$\frac{\text{m}}{\text{sec}}$	$\frac{\text{ft}}{\text{sec}}$
R	RGAS	$\frac{\ell - F}{\text{mass} - \text{Deg}}$	$2.871 \times \frac{N - m}{kg-K}$	$1.715 \times 10^3 \frac{\text{ft-lbf}}{\text{slug-R}}$
μ_∞	MUINF	$\frac{\text{mass}}{\ell - t}$	$\frac{\text{kg}}{\text{m} - \text{sec}}$	$\frac{\text{slugs}}{\text{ft} - \text{sec}}$
h_∞	HINF	$\frac{\ell^2}{t^2}$	$\frac{\text{m}^2}{\text{sec}^2}$	$\frac{\text{ft}^2}{\text{sec}^2}$
c_{p_∞}	CPINF	$\frac{\ell^2}{t^2 \text{Deg}}$	$\frac{\text{m}^2}{\text{sec}^2\text{-K}}$	$\frac{\text{ft}^2}{\text{sec}^2\text{-R}}$

Table 3. Summary of Reference Condition Input Options

Method	Required Input Values ($READ3)
Altitude	MINF; PINF or PZERO; TINF or TZERO
Wind Tunnel	MINF; PINF or PZERO; REPL
Primitive	UINF; PINF; TINF

Notes: (1) If LAMVIS = 1, the reference viscosity MUINF must be input.
(2) The reference length CLENG must always be specified.
(3) The reference molecular weight, WTREF, and the specific heat ratio, GAMMA, may be input if the default values of 28.966 and 1.4, respectively, are not desired.

Table 4 - Options for Geometry Print IGPRT(IV)

IV	VARIABLE
1	$J \frac{\partial y^1}{\partial x^1}$
2	$J \frac{\partial y^2}{\partial x^1}$
3	$J \frac{\partial y^3}{\partial x^1}$
4	$J \frac{\partial y^1}{\partial x^2}$
5	$J \frac{\partial y^2}{\partial x^2}$
6	$J \frac{\partial y^3}{\partial x^2}$
7	$J \frac{\partial y^1}{\partial x^3}$
8	$J \frac{\partial y^2}{\partial x^3}$
9	$J \frac{\partial y^3}{\partial x^3}$
10	$J \frac{\partial y^1}{\partial t}$
11	$J \frac{\partial y^2}{\partial t}$
12	$J \frac{\partial y^3}{\partial t}$
13	r_1
14	r_2
15	r_3
16	r_3^{-1}

Table 4 - Options for Geometry Print IGPRT(IV) (Continued)

17	J_1
18	J_2
19	J_3
20	J_3^{-1}
21	x_1
22	y_1
23	z_1

The subscripts on the r's and J's refer to the time level 1 = n, 2 = n + 1, 3 = n + β. All other variables are assumed to be at level n + β.

Table 5 - Plot File Variables - IVARPL(IP)

IP	Variables
1	u_1 velocity component
2	u_2 velocity component
3	u_3 velocity component
4	density
5	enthalpy (static or stagnation)
6	turbulence kinetic energy
7	turbulence dissipation rate
8	species mass fraction
•	species mass fraction
•	species mass fraction
•	species mass fraction
NEQS	species mass fraction
10	not used
26	pressure
27	temperature

Table 6 - Options for Dependent and Derived Variable Print - IVARPR(I)

I	VARIABLE
1	u - direction 1 velocity component
2	v - direction 2 velocity component
3	w - direction 3 velocity component
4	ρ - density
5	h or h_0 - static enthalpy or stagnation enthalpy
6	k - turbulent kinetic energy
7	ϵ - dissipation rate of turbulent kinetic energy
8	species - mass fraction
.	species - mass fraction
.	species - mass fraction
.	species - mass fraction
25	species - mass fraction
26	p - pressure
27	T - temperature
28	μ_ℓ - laminar viscosity
29	κ - thermal conductivity
30	c_p - specific heat
31	M - molecular weight
32	μ_T - turbulent viscosity
33	μ_{eff} - effective viscosity
34	ℓ_m - mixing length
35	$\nabla \cdot \mathbf{V}$ - divergence of V
36	D:D - dissipation
37	M - Mach number
38	T_o - stagnation temperature
39	p_o - stagnation pressure
40	C_p - pressure coefficient
41	C_{po} - stagnation pressure coefficient
42	DISTN - normal distance to nearest wall
43	IDISTN - normal distance integer
44	α - void fraction (2-phase)
45	Mass injection rate (2-phase)
46	x - momentum interchange (2-phase)
47	y - momentum interchange (2-phase)
48	z - momentum interchange (2-phase)
49	Energy interchange (2-phase)

TABLE 7 - Namelist Ordering

NL	Namelist	NLWR(NL) Default
-	$CONTROL	-
1	$READ1	1
2	$READ2	1
3	$READ3	1
4	$READ4	1
5	$READ5	1
6	$READ6	0
7	$MISC	0
8	$CHEM	0
9	$TWOP	0
10	$M864	0
11	Not used	0
12	Not used	0
--	$JANAFNL (optional)	-

Table 8 - Boundary Condition Options - IEQBC(KSURF,IADI,IEQ)

IEQBC	Boundary Condition	Required Input
	FUNCTION CONDITIONS	
1	$\Delta\phi = 0$	None
2	$\phi = 0$	None
3	$\Delta P = 0$	None
4	P = PAMB(KSURF,IADI)	PAMB(KSURF,IADI)
5	P = P(t)	NPRES1,NPRES2,PRES1, PRES2
6	$\Delta P_0 = 0$	None
7	P_0 = PTOT	PTOT
8	Not Used	
9	$\Delta T = 0$	None
10	T = TWALL(KSURF,IADI)	TWALL(KSURF,IADI)
11	T = T(t)	NTWALL1,NTWALL2,TWALL1, TWALL2
12	$\Delta H_0 = 0$	None
13	h_0 = HTOT	HTOT
14	P_0 = PTOT (Incompressible)	PTOT
15	T_0 = TTOT	TTOT
16	T_0 = TWALL(KSURF,IADI)	TWALL(KSURF,IADI)
17	P = PRESS1(IG1,IG2,IBC)	PRESS1(IG1,IG2,IBC) set in PRESSET
18	P_0 = PRESS1(IG1,IG2,IBC)	PRESS1(IG1,IG2,IBC) set in PRESSET
19	Two layer model	PRESS1,BLPROF
30	k inlet condition	AFADD
31	ϵ inlet condition	AFADD
32	Specified rotational speed	NVROT1,NVROT2,VROT1,VROT2
33	Specified flow angle	ALPINF

Table 8 - Boundary Condition Options (Continued)

IEQBC	Boundary Condition	Required Input
34	$\omega_{IEQ} = \omega_{IEQ}(t)$; mass fraction	INJT1, INJT2, INJMF1, INJMF2
35	$x_{IEQ} = x_{IEQ}(t)$; mole fraction	INJT1,INJT2,INJMF1,INJMF2
40	Porous wall boundary	PPL,PERM,TRON,DTRON,TROFF, DTROFF
42	Inlet radial velocity profile with wall injection: u_{IEQ} = UWALL•RADPRF(IG1,IG2)	UWALL,RADPRF(IG1,IG2)
70	Moving surface condition: u_{IEQ} = USURF(IEQ), IEQ = 1, 2 or 3	USURF(IEQ)
71	$\vec{n}\cdot\vec{u} = 0$, $\vec{n}$ = surface normal	None
72	$\vec{t}\cdot\vec{u} = 0$, $\vec{t}$ = surface tangent	None
99	$\Delta\phi^{***} = \Delta\phi^{**} = \Delta\phi^{*}$	None
	FIRST DERIVATIVE BOUNDARY CONDITIONS	
101	$\frac{\partial\phi}{\partial y^{IADI}} = 0$	None
102	$\vec{n}\cdot\nabla\phi = 0$	None
103	$\frac{\partial P}{\partial y^{IADI}} = 0$	None
104	$\vec{n}\cdot\nabla p = 0$	None
105	$\frac{\partial T}{\partial y^{IADI}} = 0$	None
106	$\vec{n}\cdot\nabla T = 0$	None

SECOND DERIVATIVE BOUNDARY CONDITIONS

201	$\frac{\partial^2 \phi}{\partial y^{IADI^2}} = 0$	None
202	$\frac{\partial^2 P}{\partial y^{IADI^2}} = 0$	None
203	$\frac{\partial^2 T}{\partial y^{IADI^2}} = 0$	None
204	$\frac{\partial^2 h_0}{\partial y^{IADI^2}} = 0$	None

MISCELLANEOUS BOUNDARY CONDITIONS

301	Governing equation	None

Note: ϕ is the dependent variable associated with equation IEQ and $\vec{n}$ is the unit vector normal to a surface; hence, $\vec{n} \cdot \nabla$ is the normal derivative.

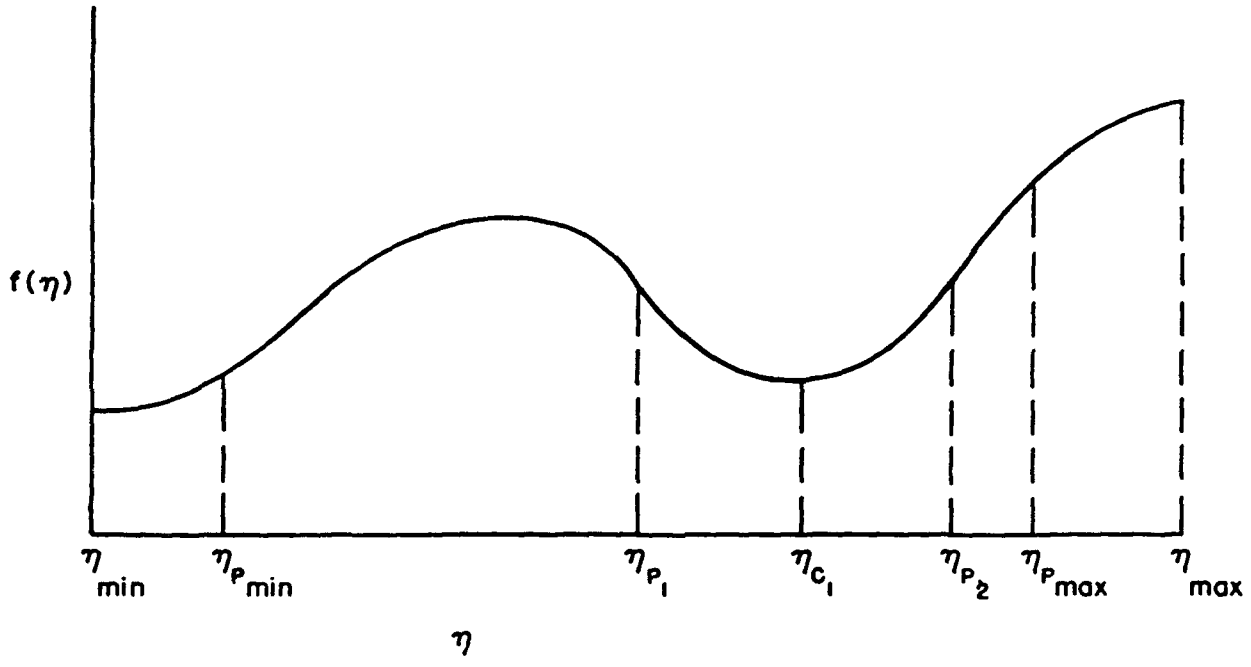


Figure 1. Oh Grid Notation.

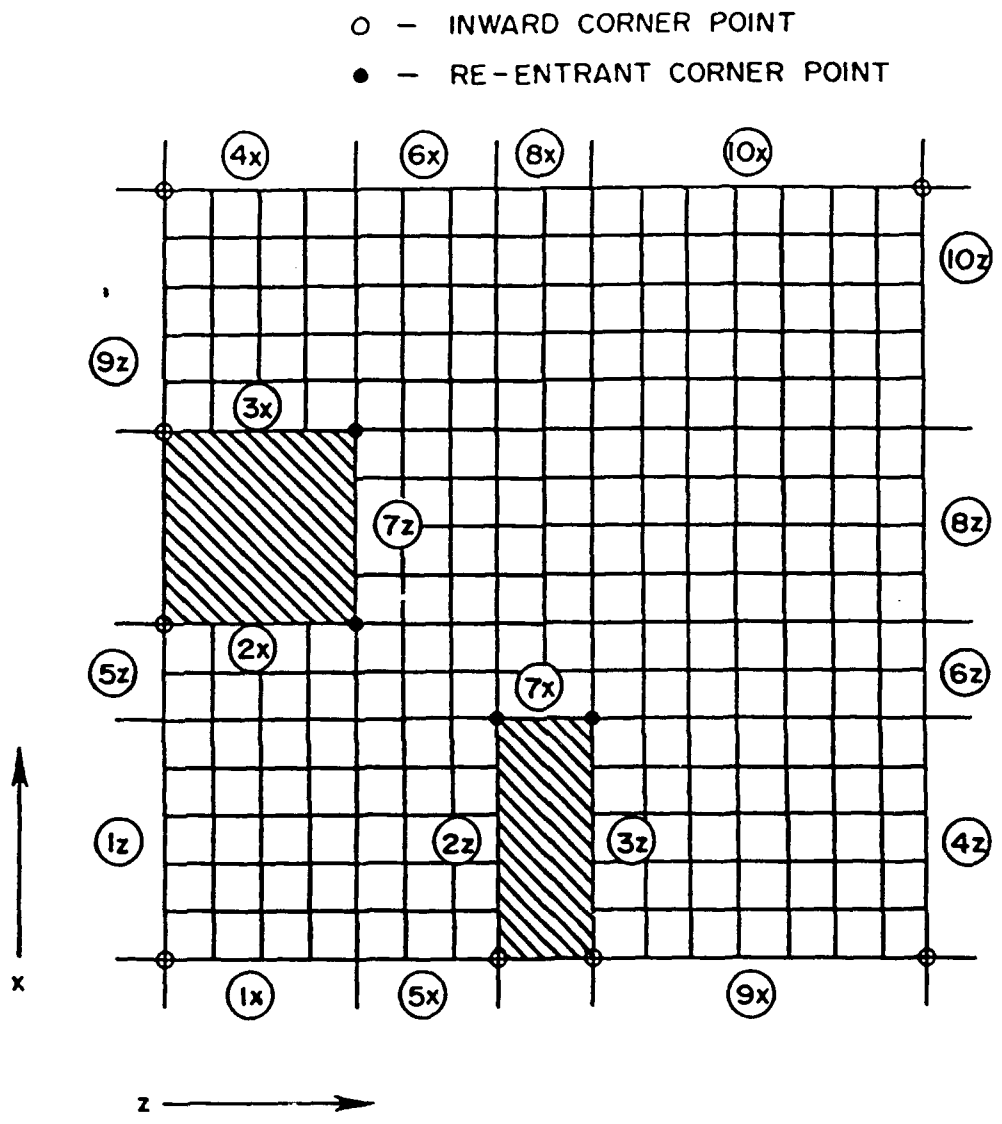


Figure 2. A Sample Computational Domain with Embedded Solid Bodies.

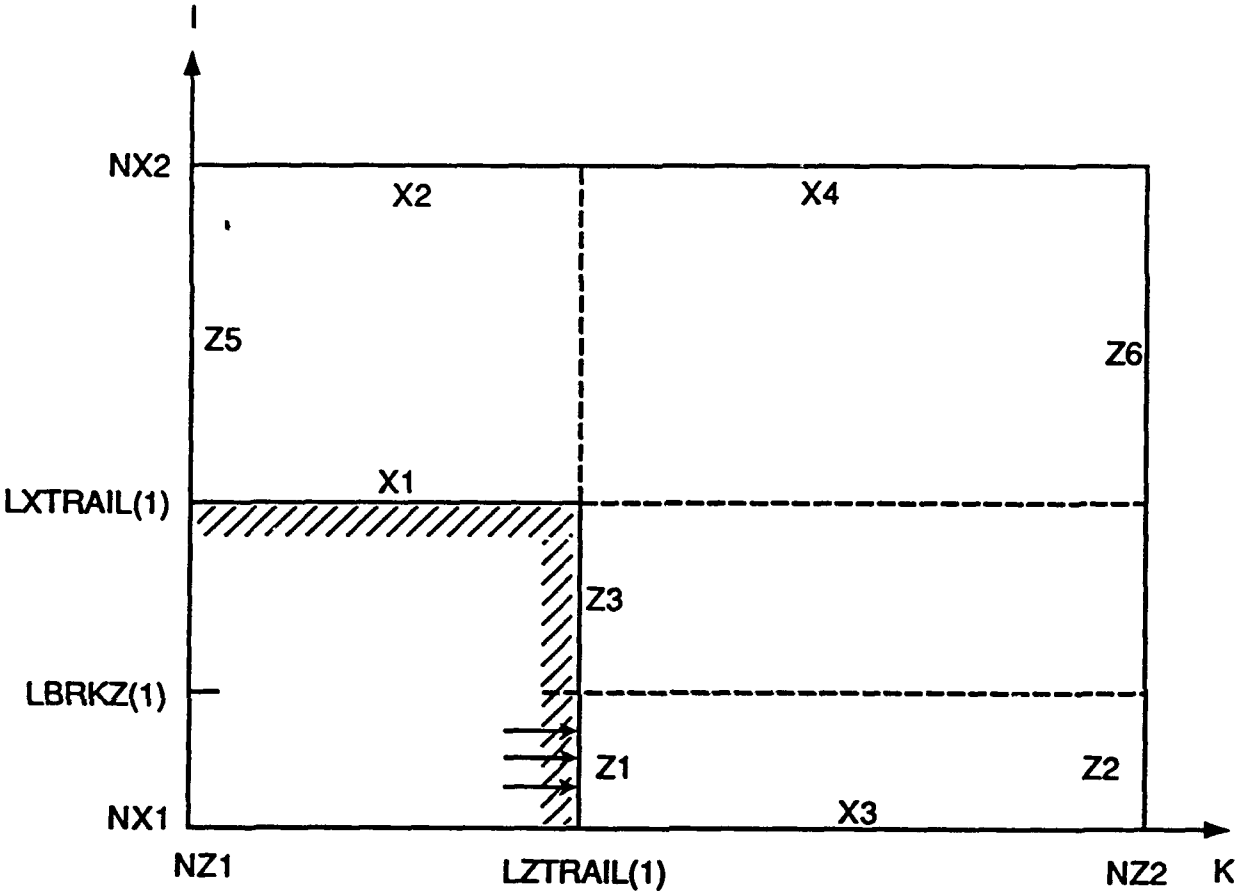


Figure 3. Nomenclature for MINT Computational Domain for Base Region Only.

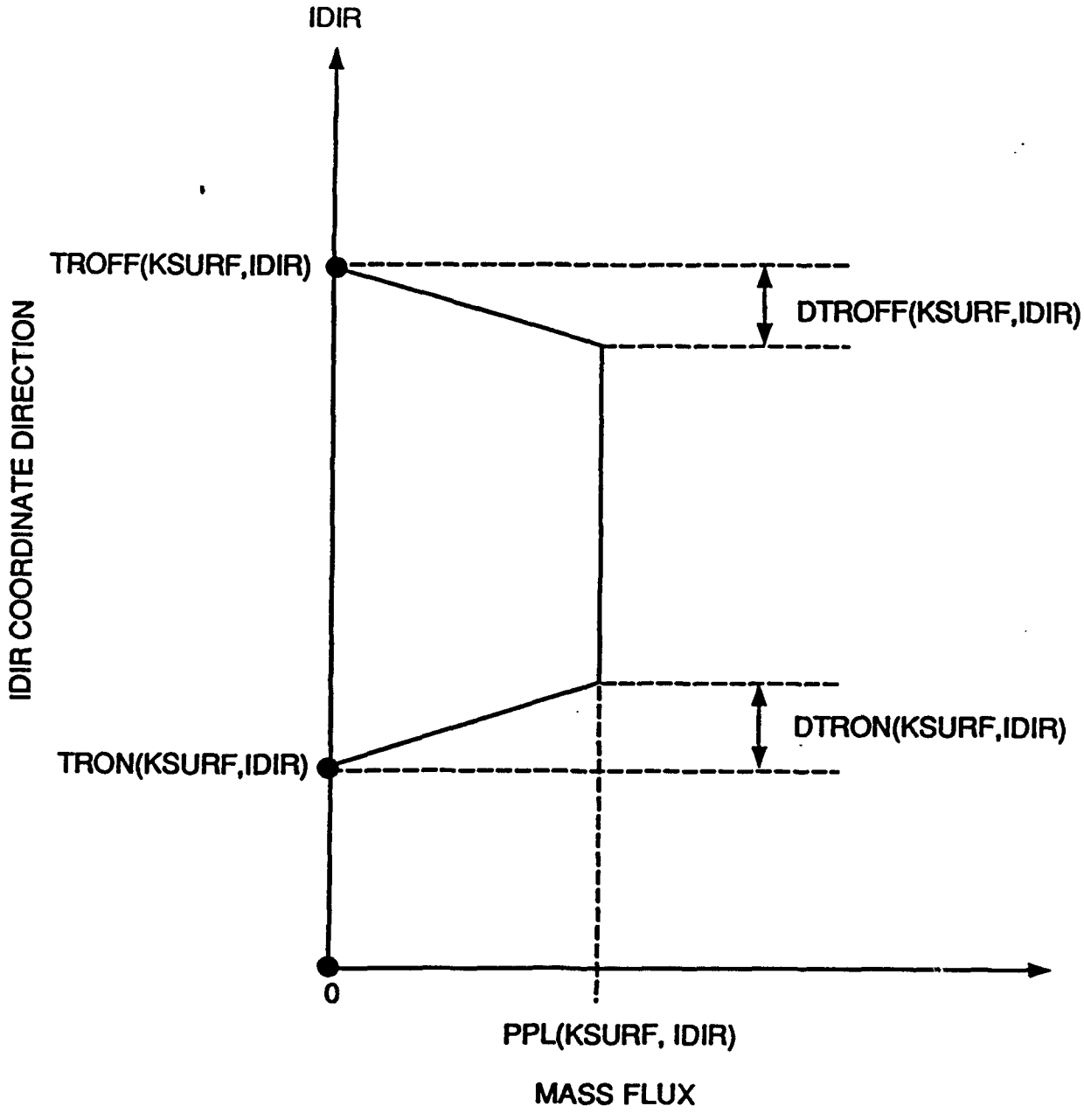
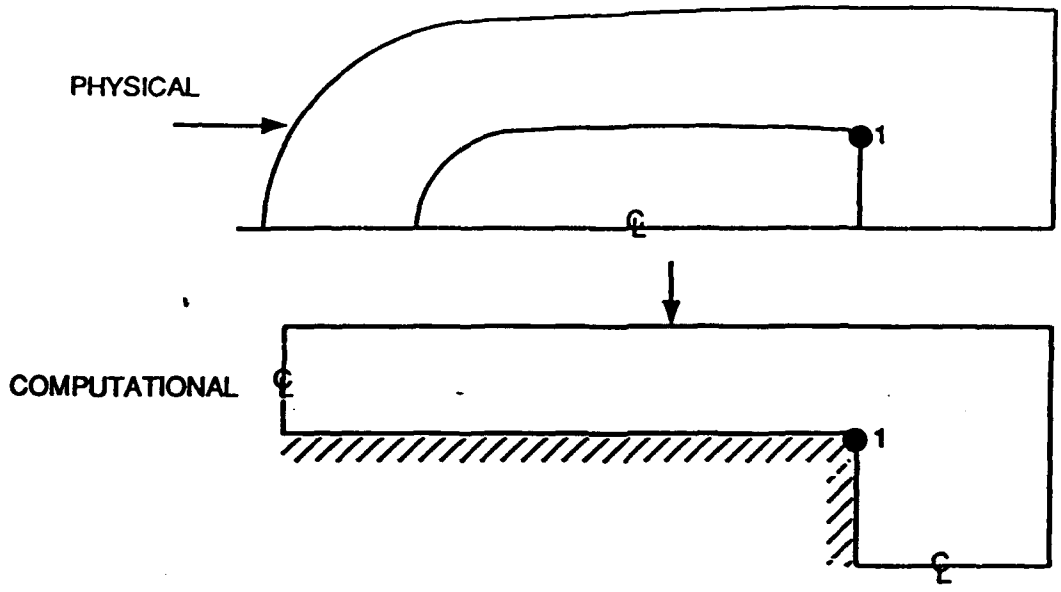
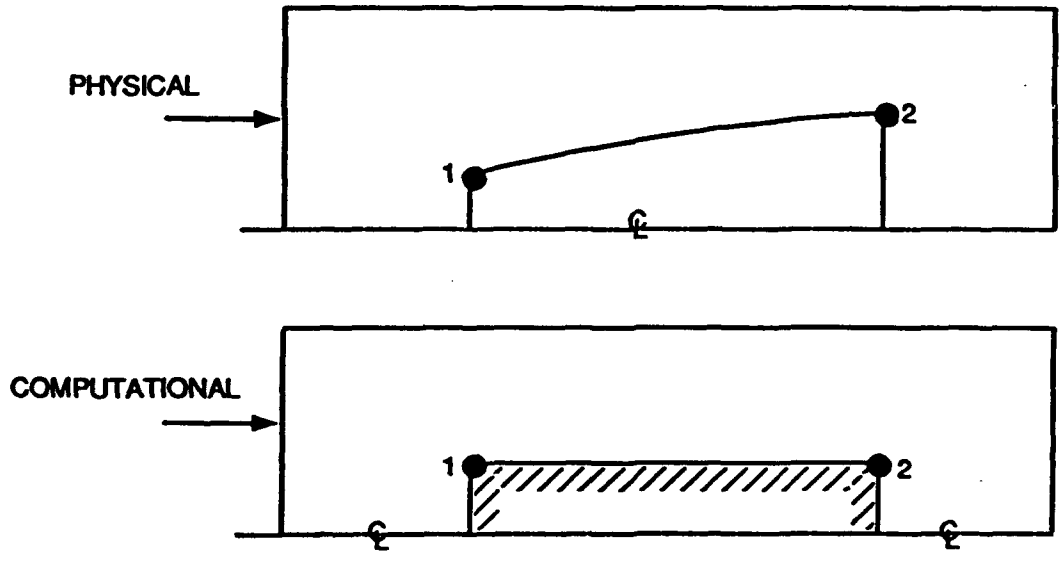


Figure 4. Nomenclature for Injection Mass Flux Boundary Condition for Direction IDIR, Surface Number KSURF and Equation IEQ: IEQBC (KSURF, IDIR, IEQ) = 40.

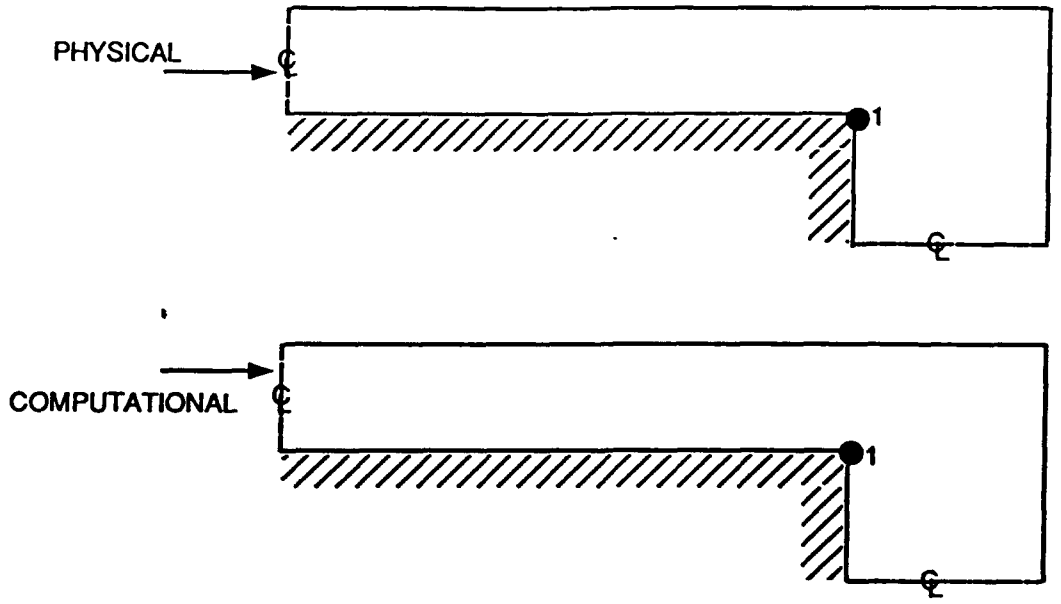


IBASE = 1, ROUND NOSE, FULL PROJECTILE

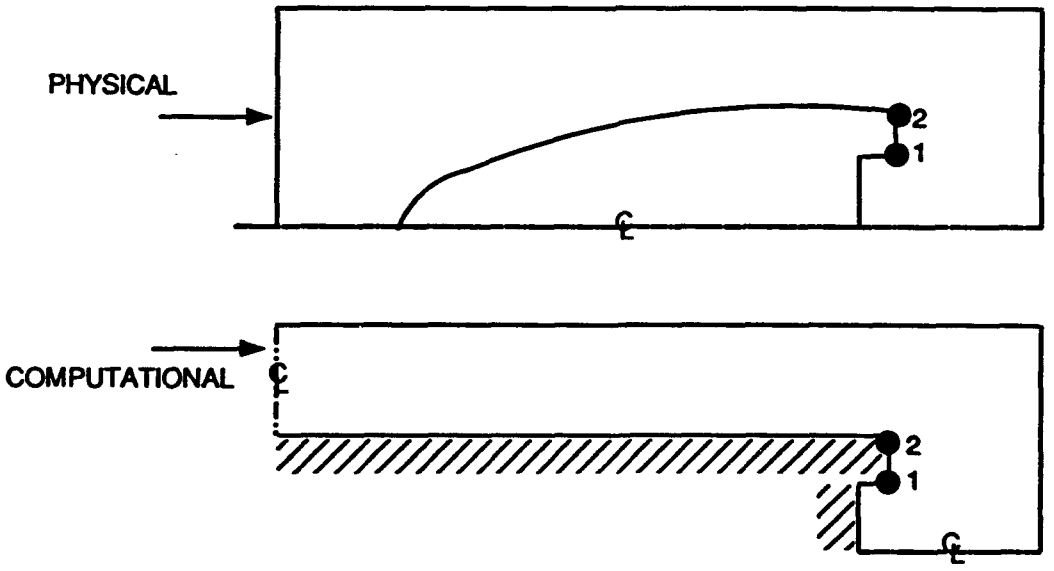


IBASE = 2, FLAT NOSE, FULL PROJECTILE

Figure 5a. Nomenclature for IBASE Input. Physical and Computational Domains.

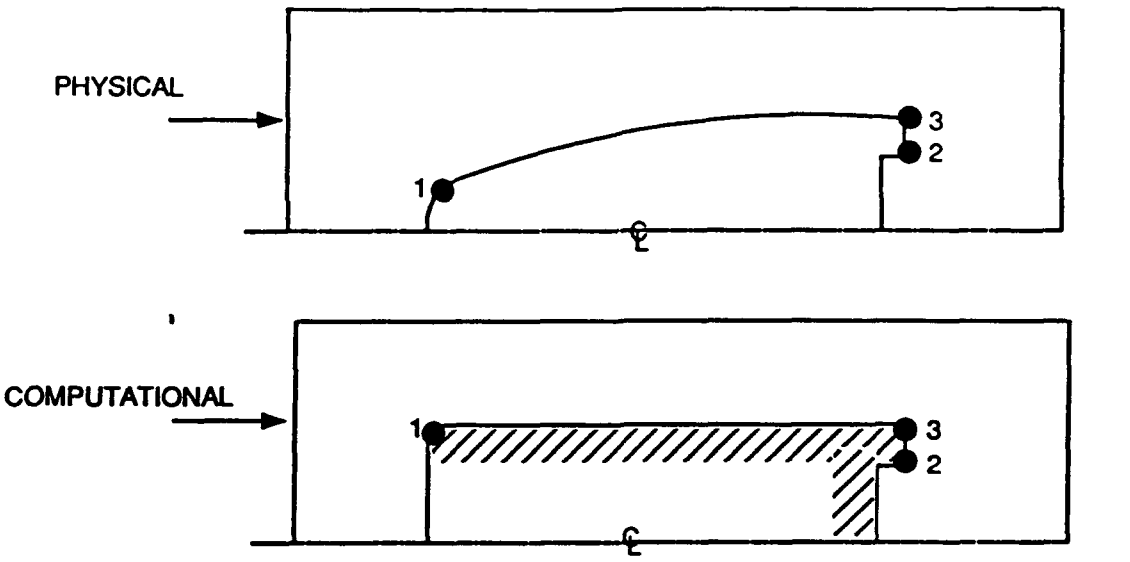


IBASE = 3, BASE ONLY

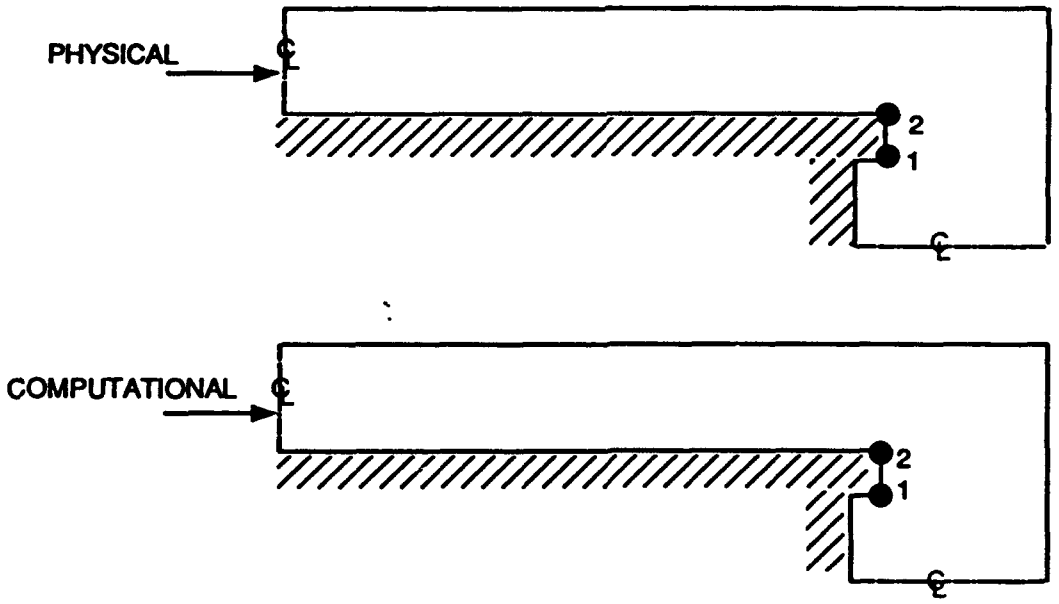


IBASE = 4, ROUND NOSE, FULL PROJECTILE, NOTCHED BASE

Figure 5b. Nomenclature for IBASE Input. Physical and Computational Domains.



IBASE = 5, FLAT NOSE, FULL PROJECTILE, NOTCHED BASE



IBASE = 6, NOTCHED BASE ONLY

Figure 5c. Nomenclature for IBASE Input. Physical and Computational Domains.

APPENDIX A

Multiple Species Gas and Chemistry Relations

Real Gas Effects

The calculation of real gas effects requires a suitable thermodynamic data base. Several thermodynamic data bases are readily available, e.g., see Kee and Miller (Ref. A-1) and Gordon and McBride (Ref. A-2). All these data bases are generated by the use of partition functions (Ref. A-3) and spectroscopic measurements. The concept of a partition allows the contributions of the various internal energy modes to be summed to calculate the total internal energy of a given molecule. Internal energy modes associated with translation, rotation, vibration and electronic levels are considered, namely the total internal energy is given by

$$e = e_{tr} + e_{rot} + e_{vib} + e_{el} \tag{A1}$$

For monatomatic molecules there is no contribution from the rotational and vibrational states, and generally the electronic contributions for all molecules are small except at higher temperatures.

Spectroscopic information has been obtained for a wide variety of molecules and curve fit constants for the calculation of the molar values of specific heat, $\hat{C}_{p_i}$, the enthalpy, $\hat{h}_i$, and the entropy, $\hat{S}_i$, are available in the form

$$\frac{\hat{C}_{p_i}(T)}{R} = a_{1i} + a_{2i}T + a_{3i}T^2 + a_{4i}T^3 + a_{5i}T^4 \tag{A2}$$

where R is the universal gas constant and i refers to the i^{th} chemical species.

The static enthalpy, $\hat{h}_i$ is defined by.

$$\hat{h}_i = \int_{T_r}^{T} \hat{C}_{p_i}\,(T')\,dT' + \hat{h}_{f_i} \tag{A3}$$

where T_r is some reference temperature (usually 298.15° K) and h_{f_i} is the heat of formation of the i^{th} species at the reference temperature. Substitution of Eq. (A2) into Eq. (A3) and integrating yields

$$\frac{\hat{h}_i}{RT} = a_{1i} + \frac{a_{2i}T}{2} + \frac{a_{3i}T^2}{3} + \frac{a_{4i}T^3}{4} + \frac{a_{5i}T^4}{5} + \frac{a_{6i}}{T} \qquad \text{(A4)}$$

where

$$a_{6_i} = \frac{\hat{h}_{f_i}}{R} - \left[a_{1i}T_r + \frac{a_{2i}T_r^2}{2} + \frac{a_{3i}T_r^3}{3} + \frac{a_{4i}T_r^4}{4} + \frac{a_{5i}T_r^5}{5}\right] \qquad \text{(A5)}$$

Likewise, for the species entropy, $\hat{S}_i$,

$$\frac{\hat{S}_i}{R} = a_{1i}\ell n\ T + a_{2i}T + \frac{a_{3i}T^2}{2} + \frac{a_{4i}T^3}{3} + \frac{a_{5i}T^4}{4} + a_{7i} \qquad \text{(A6)}$$

where

$$a_{7i} = \frac{\hat{S}_{f_i}}{R} - \left[a_{1i}\ell n\ T_r + a_{2i}T_r + \frac{a_{3i}T_r^2}{2} + \frac{a_{4i}T_r^3}{3} + \frac{a_{5i}T_r^4}{4}\right] \qquad \text{(A7)}$$

and $\hat{S}_{f_i}$ is the value of $\hat{S}_i$ at temperature T_r. Normally two sets of curve fit constants are generated for each species corresponding to low and high temperature regions. These regions are normally 300 K to 1000 K and 1000 K to 5000 K.

Within the CMINT code the thermochemical data base is established in subroutine JANAFTB. Subroutine JANAFTB is written in a general manner such that the species symbol and other necessary information is input via list directed read. The thermochemical data is then extracted from the NASA/LeRC data base file, which is generated by the CET86 code (Ref. A-2). See Section 11 for the required additional input for multiple species gases. It should be noted that an additional Namelist, $JANAFNL, is read after the list directed input to allow the user to modify any of the thermochemical data. See section 11 for a description of the variables in $JANAFNL.

Before proceeding to the discussion of the real gas initial condition, it is convenient to first define several variables. The first is the mixture frozen specific heat, $C_p(T)$,

$$C_p(T) = \sum \omega_i C_{p_i}\ (T) \qquad \text{(A8)}$$

where ω_i is the species mass fraction and the mass specific heat, C_{p_i} is related to molar specific heat, $\hat{C}_{p_i}$

$$C_{p_i} = \frac{\hat{C}_{p_i}}{M_i} \tag{A9}$$

where M_i is the species molecular weight. Substitution of Eqs. (A2) and (A9) into Eq. (A8) yields

$$C_p = \frac{R}{M}\left\{\sum \omega_i \frac{M}{M_i} a_{1i} + T \sum \omega_i \frac{M}{M_i} a_{2i} + T^2 \sum \omega_i \frac{M}{M_i} a_{3i} + T^3 \sum \omega_i \frac{M}{M_i} a_{4i} + T^4 \sum \omega_i \frac{M}{M_i} a_{5i}\right\} \tag{A10}$$

The mixture molecular weight, M, is defined by

$$\frac{1}{M} = \sum \frac{\omega_i}{M_i} \tag{A11}$$

Eq. (A10) can be written in a short hand notation as

$$C_p(T) = \frac{R}{M}\left[\tilde{A}_i + \tilde{A}_2 T + \tilde{A}_3 T^2 + \tilde{A}_4 T^3 + \tilde{A}_5 T^4\right] \tag{A12}$$

where

$$\tilde{A}_k = \sum \left[\omega_i \frac{M}{M_i} - a_{ik}\right] \tag{A13}$$

Likewise, the mixture enthalpy, h(T), can be written as

$$h(T) = \frac{R}{M}\left[\tilde{A}_i T + \frac{\tilde{A}_2 T^2}{2} + \frac{\tilde{A}_3 T^3}{3} + \frac{\tilde{A}_4 T^4}{4} + \frac{\tilde{A}_5 T^5}{5} + A_6\right] \tag{A14}$$

The isentropic stagnation pressure, P_o, can be calculated (as a function of the static pressure and the stagnation temperature) from Maxwell's relationship

$$dh = T\,ds + V\,dP \tag{A15}$$

Since the process is isentropic, ds = 0. Also

$$dh = C_p\, dT \tag{A16}$$

and

$$PV = \frac{R}{M}\, T \tag{A17}$$

Substitution of Eqs. (A16) and (A17) into Eq. (A15) yields

$$\frac{R}{M}\,\frac{C_p\, dT}{T} = \frac{dP}{P} \tag{A18}$$

By integrating

$$P_o = P \exp\left[\frac{R}{M}\int_{T_r}^{T}\frac{C_p(T')\, dT'}{T'}\right] \tag{A19}$$

For a calorically ideal gas, $C_p(T')$ is a constant and the well-known relationship is obtained

$$\frac{P_o}{P} = \left[\frac{T_o}{T}\right]^{\frac{\gamma}{\gamma-1}} \tag{20}$$

where γ is the ratio of specific heats. Substitution of Eq. (A12) into Eq. (A19) yields

$$P_o = Pe^{\frac{M}{R}\phi} \tag{A21}$$

where

$$\phi = \tilde{A}_1\, \ell n\left[\frac{T_o}{T}\right] + \tilde{A}_2[T_o - T] + \tilde{A}_3\left[\frac{T_o^2 - T^2}{2}\right] + \tilde{A}_4\left[\frac{T_o^3 - T^3}{3}\right] + \tilde{A}_5\left[\frac{(T_o^4 - T^4}{4}\right] \tag{A22}$$

Eq. (A21) for the stagnation pressure is thus seen to be a linear relationship with respect to the static pressure, P, and a transcendental relationship with respect to the static and stagnation temperatures.

The sound speed, c, is defined by the relationship

$$c^2 = \left[\frac{\partial P}{\partial \rho}\right]_s \tag{A23}$$

But for a mixture of ideal gases the gas law is

$$P = \rho \frac{R}{M} T \tag{A24}$$

and Maxwell's relationship, Eq. (A15), can be re-written as

$$dh = C_p \, dT = T \, ds \quad \frac{dP}{\rho} \tag{A25}$$

Simple manipulation of Eqs. (A23) - (A25) yields

$$c^2 = \frac{R}{M} T\left[1 - \frac{R}{MC_p}\right]^{-1} \tag{A26}$$

yielding an effective ratio of specific heats

$$\gamma = \left[1 - \frac{R}{MC_p}\right]^{-1} \tag{A27}$$

In the CMINT computer code three methods are used for determining the reference conditions, viz. the so-called altitude and wind tunnel methods, and the primitive variable method. When using the wind tunnel method the user supplies the freestream Mach number, M_∞, the free stream pressure, P_∞, and temperature, T_∞, the reference length l_∞ and the mixture molecular weight, M. From these five parameters all other free stream variables can be calculated. The free stream density can be calculated from

$$\rho_\infty = \frac{P_\infty M}{RT_\infty} \tag{A28}$$

The free stream viscosity, μ_∞, may be input or may be calculated from Sutherland's law,

$$\mu_\infty = \frac{c_1 T_\infty^{3/2}}{c_2 + T_\infty} \tag{A29}$$

where c_1 and c_2 are known constants. (See Section 10; LAMVIS, in $READ3).

The ratio of specific heats, γ, is calculated from

$$\gamma = \left[1 - \frac{R}{MC_p(T_\infty)}\right]^{-1} \tag{A30}$$

where $C_p(T_\infty)$ is calculated from Eq. (A12). The sound speed can be calculated from

$$c_\infty = \sqrt{\gamma(T_\infty)RT_\infty} \tag{A31}$$

and the free stream velocity can be calculated from

$$U_\infty = M_\infty c_\infty \tag{A32}$$

The Reynolds number can be calculated from

$$Re = \frac{\rho_\infty U_\infty \ell_\infty}{\mu_\infty} \tag{A33}$$

The free stream stagnation enthalpy h_{0_∞} is defined as

$$h_{0_\infty} = h_\infty + \frac{U_\infty^2}{2} \tag{A34}$$

or

$$h_{0_\infty} = \frac{R}{M}\left[\tilde{A}_i T_\infty + \tilde{A}_2 \frac{T_\infty^2}{2} + \tilde{A}_3 \frac{T_\infty^3}{3} + \tilde{A}_4 \frac{T_\infty^4}{4} + \tilde{A}_5 \frac{T_\infty^5}{5} + \tilde{A}_6\right] + \frac{U_\infty^2}{2} \tag{A35}$$

But the free stream stagnation temperature T_{0_∞} can also be used to define h_{0_∞} viz.

$$h_{0_\infty} = \frac{R}{M}\left[\tilde{A}_1 T_{0_\infty} + \tilde{A}_2 \frac{T_{0_\infty}^{\,2}}{2} + \tilde{A}_3 \frac{T_{0_\infty}^{\,3}}{3} + \tilde{A}_4 \frac{T_{0_\infty}^{\,4}}{4} + \tilde{A}_5 \frac{T_{0_\infty}^{\,5}}{5} + \tilde{A}_6\right] \quad \text{(A36)}$$

Combination of Eqs. (A35) and (A36) yields a transcendental relationship for T_{0_∞} as a function of T_∞ and U_∞, both known values at this point. Eq. (A36) is easily solved by the Newton-Raphson procedure to yield T_{0_∞}. Eq. (A21) can then be solved to yield P_{0_∞}.

When the wind tunnel parameters of Reynolds number per unit length Re/ℓ_∞, Mach number, M_∞, and static pressure P_∞ or stagnation pressure, P_{0_∞} are given, the procedure becomes slightly more complex. If it is first assumed that P_∞ is given, it is possible to manipulate Eqs. (A28), (A31) and (A32) to yield

$$\mu_\infty \sqrt{T_\infty} = \frac{M_\infty P_\infty}{Re/\ell_\infty}\sqrt{\frac{M}{R}\gamma(T_\infty)} \quad \text{(A37)}$$

Since M_∞, P_∞ and Re/ℓ_∞ are all known, Eq. (A37) represents a transcendental equation in T_∞ and can be solved numerically by the Newton-Raphson procedure. When P_{0_∞} is specified rather than P_∞, Eq. (A21) can be substituted into Eq. (A37) to yield

$$\mu_\infty \sqrt{T_\infty} = \frac{M_\infty P_{0_\infty}}{\frac{Re}{\ell_\infty}\exp\left[\frac{M}{R}\phi\ (T_\infty)\right]}\sqrt{\frac{M}{R}\gamma(T_\infty)} \quad \text{(A38)}$$

which is again a transcendental equation in T_∞ which can readily be solved. Once T_∞ is known, relationships similar to those used for the altitude input can be used to calculate all remaining free stream variables. Thus a consistent set of free stream relationships are available and have been incorporated into the CMINT computer code.

Finite Rate Chemistry

In general chemical reactions are of the form

$$\sum_{s=1}^{\ell} \nu'_{s,r} X_s \underset{k_{b,r}}{\overset{k_{f,r}}{\rightleftharpoons}} \sum_{s=1}^{\ell} \nu''_{s,r} X_s \quad \text{(A39)}$$

where $\nu'_{s,r}$ $\nu''_{s,r}$ are the stoichiometric coefficients appearing on the left and right of the reaction r, $k_{f,r}$ and $k_{b,r}$ are the forward and backward rate constants and $[X_s]$ is the concentration of any species X_s. It can be shown (e.g. Ref. A-2) that the rate of production of species s in reaction r is given by

$$\left[\frac{d[X_s]}{dt}\right]_r = \left[\nu''_{s,r} - \nu'_{s,r}\right] k_{f,r} \prod_{\rho=1}^{\ell} [X_\rho]^{\nu'_{\rho,r}}$$

$$+ \left[\nu'_{s,r} - \nu''_{s,r}\right] k_{b,r} \prod_{\rho=1}^{\ell} [X_\rho]^{\nu''_{\rho,r}} \qquad \text{(A40)}$$

where the forward and backward reaction rates are related by the equilibrium constant, $K_{c,r}$, viz.,

$$K_{c,r} = \frac{k_{f,r}}{k_{b,r}} \qquad \text{(A41)}$$

Usually the forward reaction rate is of the Arrhenius form

$$k_{f,r} = A_r T^{n_r} e^{-B_r/T} \qquad \text{(A42)}$$

where A_r, n_r, and B_r are constants for a given reaction r.

The equilibrium constant is actually a function of temperature and is given by the relationship

$$K_{c,r} = \frac{\exp\left[-\dfrac{\sum \left[\nu''_{s,r} - \nu'_{s,r}\right]\hat{\mu}^0_s}{RT}\right]}{RT^{\sum \left[\nu''_{s,r} - \nu'_{s,r}\right]}} \qquad \text{(A43)}$$

where R is the universal gas constant and $\hat{\mu}^0_s$ is given by

$$\hat{\mu}^0_s = \int_{T_0}^{T} \hat{c}_{p_s}\, dT + \hat{h}_{s_0} - T\left[\int_{T_0}^{T} \frac{\hat{c}_{p_s}}{T}\, dT + \hat{s}_{s_0}\right] \qquad \text{(A44)}$$

$$- RT\, \ell n\, p_0$$

where $\hat{h}_{s_0}$ and $\hat{S}_{s_0}$ are the molal enthalpy and entropy at the reference conditions T_0 and p_0. Eqs. (A41) - (A44) enable the calculation of both rate constants. Eq. (A40) represents the rate source term for each species. Since the species equations are written with the mass fractions as the dependent variables the concentrations are related to the mass fraction by

$$[X_s] = \frac{\rho \omega_s}{M_{w_s}} \tag{A45}$$

where ω_s is the mass fraction and M_{w_s} is the molecular weight of species s. Thus the source terms due to both forward and backward reactions can be expressed in terms of the dependent variables of the three velocity components, the density, the static enthalpy, and the mass fraction, and can be appropriately linearized for implicit treatment. The user needs only to input the stoichiometric coefficients and the reaction rate constants. There is also an option to allow any reaction to proceed in only the forward direction if the user desires.

For completeness the following energy conservation equation formulation is presented. The law of conservation of energy for the fluid within a volume element is given by (Ref. A-4):

$$\left\{\begin{array}{l}\text{rate of}\\ \text{accumulation}\\ \text{of internal}\\ \text{and kinetic}\\ \text{energy}\end{array}\right\} = \left\{\begin{array}{l}\text{rate of}\\ \text{internal and}\\ \text{kinetic energy}\\ \text{in}\\ \text{by convection}\end{array}\right\} - \left\{\begin{array}{l}\text{rate of}\\ \text{internal and}\\ \text{kinetic energy}\\ \text{out}\\ \text{by convection}\end{array}\right\}$$

$$+ \left\{\begin{array}{l}\text{net rate of}\\ \text{energy addition}\\ \text{by diffusive}\\ \text{processes}\end{array}\right\} - \left\{\begin{array}{l}\text{net rate of}\\ \text{work done by}\\ \text{system on}\\ \text{surroundings}\end{array}\right\} \tag{A46}$$

In this discussion the diffusion processes are restricted to the conductive energy flux, $q^{(c)}$, and the interdiffusional energy flux, $q^{(d)}$. The effects of energy transfer due to radiation, as well as external forces, are neglected. Eq. (A46) can be written in vector-tensor notation as (Ref. A-4).

$$\frac{\partial}{\partial t}\left[\rho\left[e + \frac{\mathbf{V}\cdot\mathbf{V}}{2}\right]\right] = -\nabla\cdot\left[\rho\mathbf{V}\left[e + \frac{\mathbf{V}\cdot\mathbf{V}}{2}\right]\right] - \nabla\cdot\mathbf{q}$$

$$- \nabla\cdot(p\mathbf{V}) + \nabla\cdot(\tau\cdot\mathbf{V}) \qquad \text{(A47)}$$

where $\nabla\cdot(\rho V)$ and $\nabla\cdot(\tau\cdot V)$ are the rates of work done on the fluid by pressure and viscous forces, respectively, e is the internal energy, V is the velocity vector, ρ is the density, p is the pressure, τ is the stress tensor and

$$\mathbf{q} = \mathbf{q}^{(c)} + \mathbf{q}^{(d)} \qquad \text{(A48)}$$

The static enthalpy, h, is defined as

$$h = e + \frac{p}{\rho} \qquad \text{(A49)}$$

Solution of Eq. (A49) for e and substitution into Eq. (A47) yields

$$\frac{\partial}{\partial t}\left[\rho\left[h + \frac{\mathbf{V}\cdot\mathbf{V}}{2}\right]\right] - \frac{\partial p}{\partial t} = -\nabla\cdot\left[\rho\mathbf{V}\left[h + \frac{\mathbf{V}\cdot\mathbf{V}}{2}\right]\right] - \nabla\cdot\mathbf{q} + \nabla\cdot(\tau\cdot\mathbf{V}) \qquad \text{(A50)}$$

It is to be noted that the pressure work terms are removed from the right hand side of the equation while a temporal pressure derivative is added to the left hand side of the equation.

Defining the stagnation enthalpy, h_o, by

$$h_o = h + \frac{\mathbf{V}\cdot\mathbf{V}}{2} \qquad \text{(A51)}$$

and substituting into Eq. (A50) yields

$$\frac{\partial}{\partial t}(\rho h_o) - \frac{\partial p}{\partial t} = -\nabla\cdot(\rho\mathbf{V}h_o) - \nabla\cdot\mathbf{q} + \nabla\cdot(\tau\cdot\mathbf{V}) \qquad \text{(A52)}$$

For the case of steady state flow where heat conduction and stress work can be neglected, and the stagnation enthalpy on the boundaries is either constant at a value of HTOT or a derivative condition, the solution to Eq. (A52) becomes

$$h_o = \text{HTOT} \qquad \text{(A53)}$$

Alternate forms of Eq. (A50) are available. A commonly used technique is to dot the velocity vector into the momentum equation, viz.,

$$\mathbf{V}\cdot\frac{\partial}{\partial t}(\rho\mathbf{V}) = -\mathbf{V}\cdot(\nabla\cdot\rho\mathbf{V}\mathbf{V}) + \mathbf{V}\cdot(\nabla\cdot\tau) - \mathbf{V}\cdot\nabla p \tag{A54}$$

Using the vector identities

$$\mathbf{V}\cdot\frac{\partial}{\partial t}(\rho\mathbf{V}) = \frac{\partial}{\partial t}\left[\rho\,\frac{\mathbf{V}\cdot\mathbf{V}}{2}\right] + \frac{\mathbf{V}\cdot\mathbf{V}}{2}\,\frac{\partial\rho}{\partial t} \tag{A55}$$

$$\mathbf{V}\cdot(\nabla\cdot\rho\mathbf{V}\mathbf{V}) = \nabla\cdot\left[\rho\,\frac{\mathbf{V}\cdot\mathbf{V}}{2}\,\mathbf{V}\right] + \frac{\mathbf{V}\cdot\mathbf{V}}{2}\,\nabla\cdot\rho\mathbf{V} \tag{A56}$$

and the continuity equation

$$\frac{\partial\rho}{\partial t} = -\nabla\cdot\rho\mathbf{V} \tag{A57}$$

yields

$$\frac{\partial}{\partial t}\left[\rho\,\frac{\mathbf{V}\cdot\mathbf{V}}{2}\right] = -\nabla\cdot\left[\rho\,\frac{\mathbf{V}\cdot\mathbf{V}}{2}\right] - \mathbf{V}\cdot\nabla p + \mathbf{V}\cdot(\nabla\cdot\tau) \tag{A58}$$

Subtracting Eq. (A58) from Eq. (A50) yields

$$\frac{\partial}{\partial t}(\rho h) - \frac{\partial p}{\partial t} = -\nabla\cdot(\rho h\mathbf{V}) - \nabla\cdot\mathbf{q} + \mathbf{V}\cdot\nabla P - \mathbf{V}\cdot(\nabla\cdot\tau) + \nabla\cdot(\tau\cdot\mathbf{V}) \tag{A59}$$

but

$$\nabla\cdot(\tau\cdot\mathbf{V}) = \mathbf{V}\cdot(\nabla\cdot\tau) + \tau:\nabla\mathbf{V} \tag{A60}$$

Therefore

$$\frac{\partial}{\partial t}(\rho h) - \frac{\partial p}{\partial t} = -\nabla\cdot(\rho h\mathbf{V}) - \nabla\cdot\mathbf{q} + \mathbf{V}\cdot\nabla p + \Phi \tag{A61}$$

where $\Phi = \tau:\nabla\mathbf{V}$ and is sometimes referred to as the dissipation. Eq. (A61) is often referred to as the static enthalpy form of the energy equation while Eq. (A52) is referred to as the stagnation enthalpy of the energy equation. Both of

these options are presently coded into the CMINT computer code as is Eq. (A53), the constant stagnation enthalpy form of the energy equation.

For a multiple species variable specific heat fluid, the mixture enthalpy is defined as

$$h = \sum_{i=1}^{NS} \omega_i h_i(T) \tag{A62}$$

where ω_i is the species mass fraction, and h_i is the enthalpy of species i per unit mass and is related to the molal species enthalpy $\hat{h}_i$ (Eqs. A3, A4) by

$$h_i = \frac{\hat{h}_i}{M_i} \tag{A63}$$

where M_i is the molecular weight of species i. NS refers to the total number of species. The species enthalpy is calculated from the relationship

$$\hat{h}_i = \int_{T_r}^{T} \hat{c}_{p_i}(T')\, dT' + \hat{h}_{f_i} \tag{A64}$$

$$h = \sum_{i=1}^{NS} \omega_i \left[\int_{T_r}^{T} c_{p_i}(T')dT' + h_{fi} \right] \tag{A65}$$

It is possible to form ∇h by the use of Leibniz's formula

$$\nabla h = \sum_{i=1}^{NS} \omega_i c_{p_i}(T)\nabla T + \sum_{i=1}^{NS} h_i(T)\nabla\omega_i \tag{A66}$$

Solving Eq. (A66) for ∇T yields

$$\nabla T = \frac{1}{c_{p_f}} \left[\nabla h - \sum_{i=1}^{NS} h_i(T)\nabla\omega_i \right] \tag{A67}$$

where the frozen specific heat, c_{p_f}, is given by

$$c_{p_f} = \sum_{i=1}^{N} \omega_i c_{p_i}(T) \tag{A68}$$

Returning now to the diffusive flux vector it is possible to form a simplified form of the energy equation under restricted circumstances. The heat flux vector is usually calculated using Fourier's law

$$q^{(c)} = -(\kappa_\ell + \kappa_t)\nabla T \tag{A69}$$

where κ_ℓ and κ_t are the laminar and turbulent thermal conductivity, respectively (the subscript ℓ refers to the laminar or molecular value and the subscript t refers to the turbulent value). The interdiffusional energy flux is given by

$$q^{(d)} = \sum_{i=1}^{NS} h_i(T) j_i \tag{A70}$$

where j_i is the diffusive mass flux. When Fick's law is valid, j_i can be represented by

$$j_i = -\rho(D_\ell + D_t)\nabla\omega_i \tag{A71}$$

where D_ℓ and D_t are the laminar and turbulent diffusion coefficients, respectively. Combining Eqs. (A48), (A67), (A69), (A70) and (A71) yields

$$q = -\frac{\kappa_\ell + \kappa_t}{C_{p_f}}\left[\nabla h - \sum_{i=1}^{NS} h_i(T)\nabla\omega_i\right] - \rho\sum_{i=1}^{NS}(D_\ell + D_t)h_i(T)\nabla\omega_i \tag{A72}$$

or

$$q = -\frac{\kappa_\ell + \kappa_t}{C_{p_f}}\nabla h + \sum_{i=1}^{NS}\left[\frac{\kappa_\ell + \kappa_t}{C_{p_f}} - \rho\,(D_\ell + D_t)\right]h_i(T)\nabla\omega_i \tag{A73}$$

Now if the term within the square brackets is equal to zero, the second term of Eq. (A72) can be dropped and thus the form of the energy equation for a multiple species fluid can be considerably simplified. Using the definition for Prandtl, Pr, and Schmidt, Sc, numbers:

$$Pr = \frac{C_p\mu}{\kappa} \tag{A74}$$

$$Sc = \frac{\mu}{\rho D} \tag{A75}$$

where μ is the viscosity, then Eq. (A73) can be written as

$$q = -\left[\frac{\mu_\ell}{Pr_\ell} + \frac{\mu_t}{Pr_t}\right]\nabla h + \sum_{i=1}^{NS}\left\{\mu_\ell\left[\frac{1}{Pr_\ell} - \frac{1}{Sc_\ell}\right] + \mu_t\left[\frac{1}{Pr_t} - \frac{1}{Sc_t}\right]\right\}h_i(T)\nabla\omega_i \tag{A76}$$

The Lewis number is defined as

$$Le = \frac{Pr}{Sc} \tag{A77}$$

If the laminar and turbulent Lewis numbers are both unity then the second term of Eq. (A76) can be neglected. Hence the energy equation becomes

$$\frac{\partial}{\partial t}(\rho h) - \frac{\partial p}{\partial t} = -\nabla\cdot(\rho h V) + \nabla\cdot\left[\frac{\mu_\ell}{Pr_\ell} + \frac{\mu_T}{Pr_t}\right]\nabla h + V\cdot\nabla p + \Phi \tag{A78}$$

Eq. (A78) represents the simplified form of the energy equation for multiple species flows and the temperature linearization is considerably simplified. It is unusual that the Lewis number is identically unity, but for many flows the Lewis number is approximately unity and hence Eq. (A78) can be viewed as an approximate form of the energy equation under these conditions.

REFERENCES

A-1. Kee, R.J., Miller, J.A. and Jefferson, T.H.: "CHEMKIN: A General Purpose Problem-Independent, Transportable, Fortran Chemical Kinetics Code Package," Sandia National Laboratories Report, SAND80-8003, 1980.

A-2. Gordon, S. and McBride, B.J.: "Computer Program for Calculation of Complex Chemical Equilibrium Compositions, Rocket Performance, Incident and Reflected Shocks and Chapman-Jouguet Detonations," NASA SP-273, Interim Revision, March 1976, CET86 version obtained from B.J. McBride, NASA Lewis Research Center, Cleveland, OH.

A-3. Vincenti, W.G. and Kruger, C.H., Jr.: Introduction to Physical Gas Dynamics, John Wiley and Sons, Inc., New York, 1967.

A-4. Bird, R.B., Stewart, W.E. and Lightfoot, E.N.: Transport Phenomena, John Wiley Inc., New York, 1960, p. 313.

INTENTIONALLY LEFT BLANK

No. of Copies Organization

2 Administrator
Defense Technical Info Center
ATTN: DTIC-DDA
Cameron Station
Alexandria, VA 22304-6145

1 Commander
U.S. Army Materiel Command
ATTN: AMCAM
5001 Eisenhower Ave.
Alexandria, VA 22333-0001

1 Director
U.S. Army Research Laboratory
ATTN: AMSRL-D
2800 Powder Mill Rd.
Adelphi, MD 20783-1145

1 Director
U.S. Army Research Laboratory
ATTN: AMSRL-OP-CI-A,
Tech Publishing
2800 Powder Mill Rd.
Adelphi, MD 20783-1145

2 Commander
U.S. Army Armament Research,
Development, and Engineering Center
ATTN: SMCAR-IMI-I
Picatinny Arsenal, NJ 07806-5000

2 Commander
U.S. Army Armament Research,
Development, and Engineering Center
ATTN: SMCAR-TDC
Picatinny Arsenal, NJ 07806-5000

1 Director
Benet Weapons Laboratory
U.S. Army Armament Research,
Development, and Engineering Center
ATTN: SMCAR-CCB-TL
Watervliet, NY 12189-4050

(Unclass. only)1 Commander
U.S. Army Rock Island Arsenal
ATTN: SMCRI-TL/Technical Library
Rock Island, IL 61299-5000

1 Director
U.S. Army Aviation Research
and Technology Activity
ATTN: SAVRT-R (Library)
M/S 219-3
Ames Research Center
Moffett Field, CA 94035-1000

No. of Copies Organization

1 Commander
U.S. Army Missile Command
ATTN: AMSMI-RD-CS-R (DOC)
Redstone Arsenal, AL 35898-5010

1 Commander
U.S. Army Tank-Automotive Command
ATTN: ASQNC-TAC-DIT (Technical
Information Center)
Warren, MI 48397-5000

1 Director
U.S. Army TRADOC Analysis Command
ATTN: ATRC-WSR
White Sands Missile Range, NM 88002-5502

1 Commandant
U.S. Army Field Artillery School
ATTN: ATSF-CSI
Ft. Sill, OK 73503-5000

(Class. only)1 Commandant
U.S. Army Infantry School
ATTN: ATSH-CD (Security Mgr.)
Fort Benning, GA 31905-5660

(Unclass. only)1 Commandant
U.S. Army Infantry School
ATTN: ATSH-CD-CSO-OR
Fort Benning, GA 31905-5660

1 WL/MNOI
Eglin AFB, FL 32542-5000

Aberdeen Proving Ground

2 Dir, USAMSAA
ATTN: AMXSY-D
AMXSY-MP, H. Cohen

1 Cdr, USATECOM
ATTN: AMSTE-TC

1 Dir, ERDEC
ATTN: SCBRD-RT

1 Cdr, CBDA
ATTN: AMSCB-CI

1 Dir, USARL
ATTN: AMSRL-SL-I

10 Dir, USARL
ATTN: AMSRL-OP-CI-B (Tech Lib)

No. of
Copies Organization

4 Commander
U.S. Army Armament Research,
Development and Engineering Center
ATTN: SMCAR-AET-A,
R. Kline
H. Hudgins
S. Kahn
J. Grau
Picatinny Arsenal, NJ 07806-5001

1 Commander
U.S. Army Missile Command
ATTN: AMSMI-RD-SS-AT,
B. Walker
Redstone Arsenal, AL 35898-5010

1 Commander
U.S. Naval Surface Weapons Center
ATTN: Dr. F. Moore
Dahlgren, VA 22448

3 Commander
Naval Surface Weapons Center
ATTN: Code R44,
Dr. F. Priolo
Dr. A. Wardlaw
K24, B402-12, Dr. W. Yanta
White Oak Laboratory
Silver Spring, MD 20903-5000

1 USAF Wright Aeronautical
Laboratories
ATTN: AFWAL/FIMG,
Dr. J. Shang
WPAFB, OH 45433-6553

3 Director
National Aeronautics and Space
Administration
Langley Research Center
ATTN: Tech Library
Dr. M. J. Hemsch
Dr. J. South
Langley Station
Hampton, VA 23665

No. of
Copies Organization

3 Director
National Aeronautics and Space
Administration
Ames Research Center
ATTN: MS-227-8, L. Schiff
MS-258-1,
T. Holst
D. Chaussee
Moffett Field, CA 94035

1 Massachusetts Institute of
Technology
ATTN: Tech Library
77 Massachusetts Avenue
Cambridge, MA 02139

3 Sandia National Laboratories
ATTN: Dr. Daniel Barnette
Dr. W. Oberkampf
Dr. F. Blottner
Division 1554
P.O. Box 5800
Albuquerque, NM 87185

1 University of California, Davis
Department of Mechanical Engineering
ATTN: Prof. H. A. Dwyer
Davis, CA 95616

1 VRA Inc.
ATTN: Dr. Clark H. Lewis
P.O. Box 50
Blacksburg, VA 24060

1 University of Maryland
Department of Aerospace Engineering
ATTN: Dr. J. D. Anderson, Jr.
College Park, MD 20742

1 University of Texas
Department of Aerospace Engineering
and Engineering Mechanics
ATTN: Dr. D. S. Dolling
Austin, TX 78712-1055

1 University of Florida
Department of Engineering Sciences
College of Engineering
ATTN: Prof. C. C. Hsu
Gainesville, FL 32611

No. of

Copies Organization

1 University of Florida
Department of Engineering Sciences
College of Engineering
ATTN: Prof. C. C. Hsu
Gainesville, FL 32611

1 Pennsylvania State University
Department of Mechanical Engineering
ATTN: Dr. Kenneth Kuo
University Park, PA 16802

1 Florida Atlantic University
Department of Mechanical Engineering
ATTN: Dr. W. L. Chow
Boca Raton, FL 33431

1 Georgia Institute of Technology
School of Aerospace Engineering
ATTN: Dr. Warren C. Strahle
Atlanta, GA 30332

1 AEDC
Calspan Field Service
ATTN: Dr. John Benek
MS 600
Tullahoma, TN 37389

1 McDonnell Douglas Corporation
Dept 222 Bldg 110 Lev 1 RM/PT 151
Mail Code 5
ATTN: Dr. Thomas P. Gielda
P.O. Box 516
Saint Louis, MO 63166-0516

1 Visual Computing
ATTN: Jeffrey Q. Cordova
883 N. Shoreline Blvd.
Suite B210
Mountain View, CA 94043

1 MDA Engineering, Inc.
ATTN: John P. Steinbrenner
500 E. Border St.
Suite 401
Arlington, TX 76010

INTENTIONALLY LEFT BLANK.

USER EVALUATION SHEET/CHANGE OF ADDRESS

This Laboratory undertakes a continuing effort to improve the quality of the reports it publishes. Your comments/answers to the items/questions below will aid us in our efforts.

1. ARL Report Number ARL-CR-3 Date of Report November 1992

2. Date Report Received _____

3. Does this report satisfy a need? (Comment on purpose, related project, or other area of interest for which the report will be used.) _____

4. Specifically, how is the report being used? (Information source, design data, procedure, source of ideas, etc.) _____

5. Has the information in this report led to any quantitative savings as far as man-hours or dollars saved, operating costs avoided, or efficiencies achieved, etc? If so, please elaborate. _____

6. General Comments. What do you think should be changed to improve future reports? (Indicate changes to organization, technical content, format, etc.) _____

CURRENT ADDRESS

_____
Organization

_____
Name

_____
Street or P.O. Box No.

_____
City, State, Zip Code

7. If indicating a Change of Address or Address Correction, please provide the Current or Correct address above and the Old or Incorrect address below.

OLD ADDRESS

_____
Organization

_____
Name

_____
Street or P.O. Box No.

_____
City, State, Zip Code

(Remove this sheet, fold as indicated, staple or tape closed, and mail.)

DEPARTMENT OF THE ARMY

OFFICIAL BUSINESS

BUSINESS REPLY MAIL
FIRST CLASS PERMIT No 0001, APG, MD

Postage will be paid by addressee

Director
U.S. Army Research Laboratory
ATTN: AMSRL-OP-CI-B (Tech Lib)
Aberdeen Proving Ground, MD 21005-5066

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES